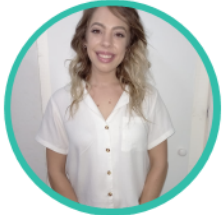




Bea Scheepers
(Team Leader)



Student No:
u16006888
Email:
u16006888@tuks.co.za
Contact No:
072 156 9991



**Stephan
Francois Bergh**



Student No:
u17000930
Email:
u17000930@tuks.co.za
Contact No:
073 770 0198



**Kyla
Foxcroft-Jones**



Student No:
u18182454
Email:
u18182454@tuks.co.za
Contact No:
076 793 4378.



Roberto Adinolfi



Student No:
u18024361
Email:
u18024361@tuks.co.za
Contact No:
071 783 5867



**Vrajesh Alkesh
Patel**



Student No:
u18102672
Email:
u18102672@tuks.co.za
Contact No:
081 489 2514



Project Proposal

The purpose of this project proposal is to introduce our client, describe the business problem they are currently facing as well as a proposed solution for this problem. This document consists out of the following sections: client information, a project request, a preliminary investigation, and problem analysis of the current versus proposed system. It also includes a requirements analysis and a feasibility/decision analysis of the proposed system.

Our Client: TutorDevOps

TutorDevOps are a South African based online coding institute completely Independent from higher learning Institutions that aims to equip students with all the fundamental tools to thrive in the software development /coding environment.



Contact Person:

Calvin Munava

Number:

081 536 6864

Email:

tutordevopsza@gmail.com



Table of Contents

Project Proposal	0
2. Client information	4
Introduction	4
2.1 History and background of the organisation.....	4
2.1.1 History	4
2.1.2 Background	5
2.2 Description of the organisation, organisational structure, area of business and business environment.....	5
2.2.1 Organisation description	5
2.2.2 Organisational structure.....	6
2.2.3 Area of business.....	7
2.2.4 Business environment.....	8
2.3 Details, position, and background of contact person	9
Conclusion.....	9
3. Project request	10
Introduction	10
3.1 Complete description of project request	10
Conclusion.....	11
4. Preliminary Investigation	12
Introduction	12
4.1 Problem/Vision Statement.....	12
4.2 Business Objectives and Goals	13
4.2.1 Analysis of our Business Goal	14
4.3 Detailed Problems, Opportunities & Directives Matrix	15
4.4 Preliminary Assumptions	17
4.5 Elicitation Techniques	18
4.7 Elicitation Technique Documentation	19
4.7.1 Interview.....	19
4.7.2 Brainstorming	22
4.7.3 Workshop	25
4.8 Details of Elicitation Tasks.....	27
4.8.1 Interview Details	27
4.8.2 Brainstorming Details.....	29
4.8.3 Workshop Details	30



Conclusion.....	31
5. Problem Analysis	32
Introduction	32
5.1 Current System Rich Picture.....	32
5.2 Current System Capability and Gap Analysis.....	33
5.3 Proposed System Rich Picture	37
5.3.1 Table for explaining proposed rich picture.....	38
Conclusion.....	39
6. Requirement Analysis	40
Introduction	40
6.1 Functional requirement list.....	40
1. Admin Subsystem	40
2. Login Subsystem.....	41
3. Student Subsystem.....	42
4. Tutor Subsystem	42
5. Payment Subsystem	44
6. Report Subsystem	44
6.2 Functional Requirement Description and Details.....	45
6.2.1 Admin Sub-system	45
6.2.2 Login Sub-system	67
6.2.3 Student Sub-system	72
6.2.4 Tutor Sub-system	86
6.2.5 Payment Sub-system.....	109
6.2.6 Report Sub-system	111
6.3 Non-Functional Requirements list.....	115
Conclusion.....	119
7. Feasibility/Decision Analysis	120
Introduction:	120
7.1 Purpose, Background, Scope and Structure:	120
7.1.1 Purpose:.....	120
7.1.2 Background.....	121
7.1.3 Scope and Structure:	121
7.2 Feasibility Analysis	123
7.2.1 Feasibility Analysis Matrix	123
7.2.2 Recommendations:	133



Conclusion:.....	135
References:.....	136
Document Conclusion.....	136
8. Appendix A: Client documentation	137
Introduction	137
8.1 Documents, Forms and Manuals	137
8.1.1 Adverts	137
8.1.2 Current Website.....	138
8.1.3 Google Forms.....	141
8.1.4 Emails	143
Conclusion	150
9. Appendix B	151
Introduction	151
9.1 Literature Study	151
9.1.1 123Tutors	151
9.1.2 TutorClass	152
9.1.3 TutorPanel	153
9.1.4 Wix	154
9.1.5 Tutorially.....	157
9.1.6 TeachWorks	160
9.1.7 Tutor Panel	164
References.....	169
10. Appendix C: Complexity	170
Introduction	170
Conclusion.....	172
11. Appendix D: Table of Figures.....	173
Client Sign-off.....	174



2. Client information

Introduction

This section will investigate the detailed information of our client, TutorDevOps (TDO), such as their history and how they came to exist. The organisational and management structure of their business will be discussed as well as their business environment and area of business. Lastly, details of our contact person (Calvin Munava), will be discussed and how he fits into the business structure.

2.1 History and background of the organisation

2.1.1 History

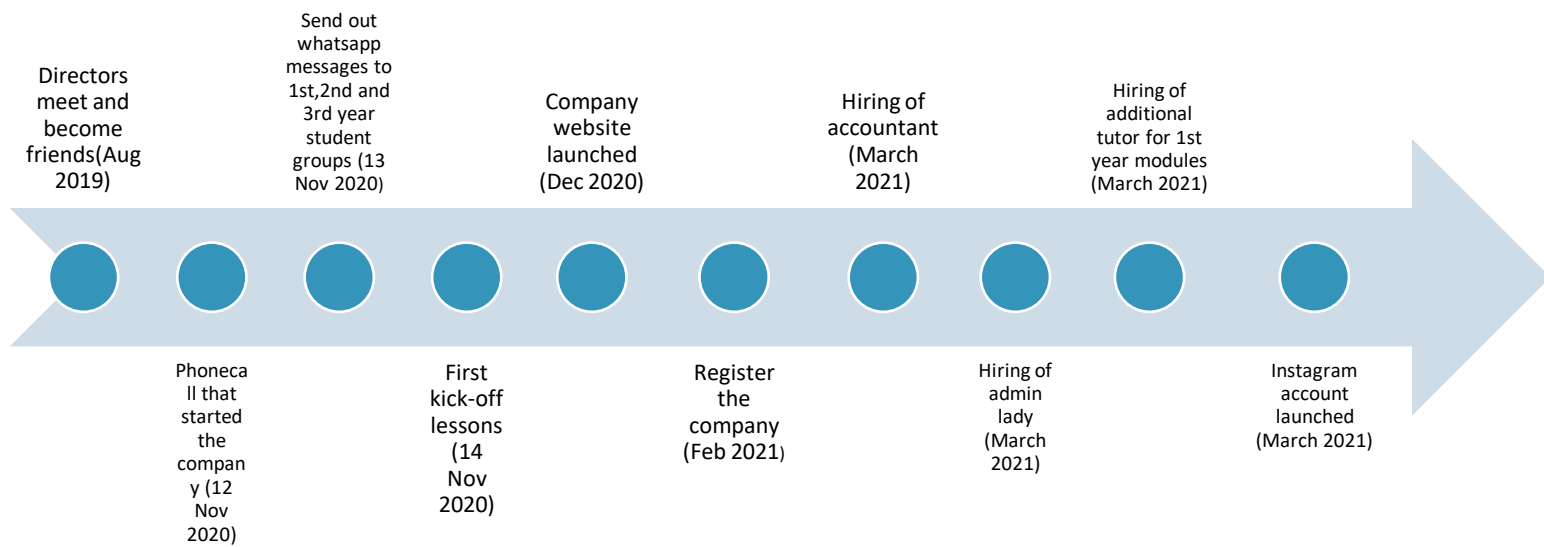
The directors (Calvin and Iain) of TDO met in August 2019 through a mutual friend who also studied Informatics and all three were completing the INF 370 module at the time. Iain and Calvin connected over shared interests in IT technology and fancy features and so their friendship continued to grow over time. Around November 2020, they shared a phone call over an Angular problem Calvin was experiencing and ended up talking about the need for tutoring in the Informatics modules. They had received multiple messages about students struggling and needing help concerning the Informatics modules. One of our group members (Bea Scheepers) also made a request to the @lifeattuks Instagram page requesting them to post a query for help with coding work. This was seen by Iain and confirmed the need for coding tutoring, growing under students.

They decided to set up their tutoring business during the above phone call and before the phone call ended, messages were already sent on WhatsApp groups for 1st, 2nd and 3rd year Informatics students offering their services. The first lessons took place that Saturday and TDO has been growing in numbers since, launching their website in December 2020. The business was then subsequently registered in February 2021.

As time passed, TDO migrated from using platforms such as google drive and discord, to others such as DropBox and Zoom. Growth within the business was seen by hiring an administrative lady, accountant, and an additional tutor during March 2021. Their Instagram page, @tutordevops, was recently launched as they continued to grow over social platforms and in numbers.



2.1.2 Background



2.2 Description of the organisation, organisational structure, area of business and business environment

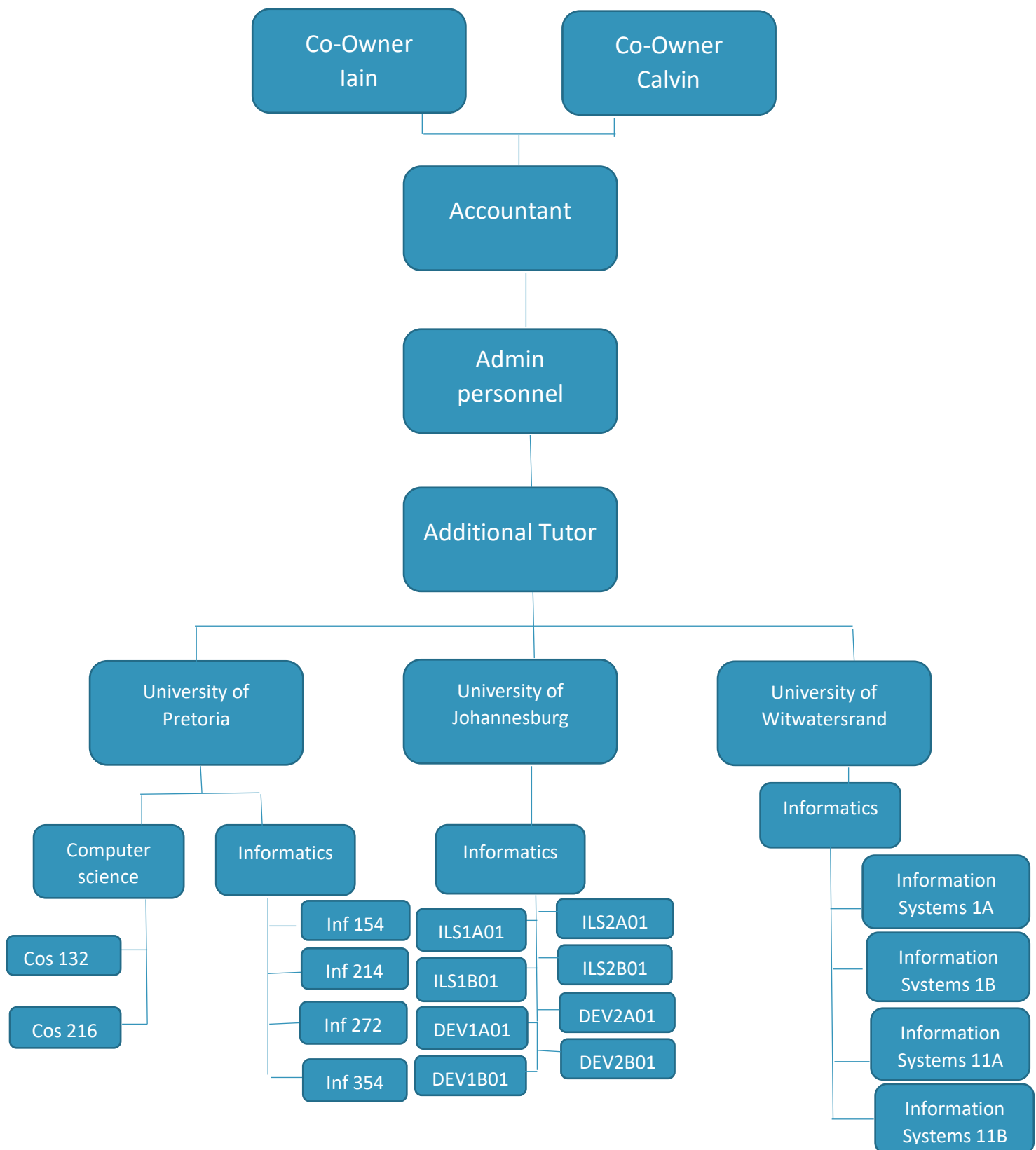
2.2.1 Organisation description

TDO is a South African based online coding institute completely Independent from higher learning Institutions, that aims to equip students with all the fundamental tools to thrive in the software development/coding environment.

Their digital presence is well known as they make use of cutting-edge technologies to deliver high quality tutoring over the digital scope.



2.2.2 Organisational structure





2.2.3 Area of business

TDO is an online institution and is seen as a virtual university. They do not have any physical premises from which they operate but are planning on locating to offices as soon as their growth allows it. They are expanding at a rapid pace and will soon be tutoring not only varsity students but also high school students with Information technology coding. The tutor classes currently cater for all coding modules of the University of Pretoria, the University of Johannesburg, and the University of Witwatersrand. TDO is however looking at offering their services to students from other universities and therefore expanding the business.

Below is a picture of their logo:



Figure 1 - TDO Logo

Below is a map of all the universities in South Africa. The three with rectangles around them indicate the universities with coding modules that TDO currently offers tutoring for, thus it can be seen that their services are offered nationally:

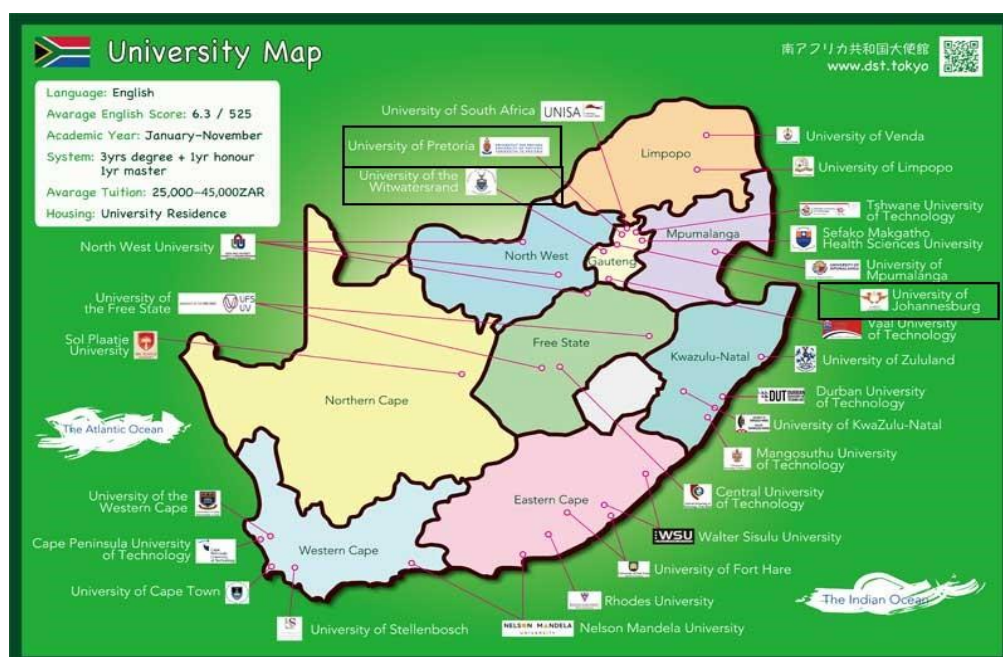


Figure 2- Map of SA Universities



2.2.4 Business environment

TDO makes use of the following technologies to boost student's confidence in their coding abilities:

- Angular
- JSON
- jQuery
- JavaScript
- ASP.NET Core
- CSS
- TypeScript
- AJAX
- ASP.NET
- Ionic
- HTML
- .NET MVC
- Ubuntu
- Microsoft SQL Server
- Python
- Java
- C++

TDO's ability to offer tutoring in all the above coding languages gives them a competitive advantage in the online tutoring environment.

The fact that they are Informatics alumni from the University of Pretoria truly sets them apart from other tutoring companies as they can associate with the students in a personal manner and thus are able to offer a service of a high standard.



2.3 Details, position, and background of contact person

The contact person at TDO is one of the directors/founders, Calvin Munava. Mr Munava is an athlete for the University of Pretoria and is currently training to take part in the Olympics for track and field. He is working fulltime as a Software Developer at a company based in Pretoria and runs TDO with his partner Iain, as a side business. He has over 4 years of experience in tutoring students in coding modules. His two big passions are athletics and his love for mastering coding, leading him to guide those that have the same passion, through their company TDO.



Email: tutordevopsza@gmail.com



Contact number: 081 536 6864

Figure 3 - Calvin Munava, Owner of TDO

Conclusion

From the section above it is seen that TDO is an online institution that grew from the vision of two friends, into a fast-growing company. This growth can be seen through the business meeting the demand for high school tutoring for coding material, as well as expanding into other universities. TDO makes use of a variety of technologies, equipping students with knowledge and skills to master software development/coding.

3. Project request

Introduction

The following section provides information regarding the need for a system by our client as well as specific details on what the system is required to do. The project request identifies the current systems capabilities as well as the proposed systems capabilities and investigates shortfalls needing to be addressed.

3.1 Complete description of project request

Currently students register on TDO's Wix Static website, after purchasing a package on the website students receive an email stating further joining information. Students are added to WhatsApp groups and GitHub repos. TDO directors manually generate communication messages across the abovementioned WhatsApp groups. Group session times are voted for by students over WhatsApp whereafter the tutor manually counts votes and sends a confirmation message regarding session times for the week. Zoom links for the sessions are sent out manually by the tutor as well as a reminder message of the scheduled session. All tutor sessions are recorded for students that could not attend the live session, these recordings are uploaded to DropBox and links to view these videos are sent to students via email. After the payment period comes to an end the directors can see who did not make payments for the upcoming period and send them a reminder invoice via email or remove them from the WhatsApp groups and GitHub repos if necessary. If a student selects an individual package, private messages are sent to arrange session times between the tutor and student. From the above it can be seen that the current system used by TDO allows most features to be done manually, especially communication.

The system TDO wants us to develop will allow all the above functions to be completed on the system itself and will eliminate the use of third-party applications. The use of these third-party applications (Google drive, GitHub, Discord/Zoom) are inefficient and thus the system requires the integration of the services offered by these applications. TDO would like the system to cater for automated responses, for example an automated welcome email with additional information to students after they register on the website. These emails will contain information regarding session bookings, communication channels, etc. Tutors will be able to schedule sessions on the system which students will vote for considering the timeslot that suits them best. After the voting polls reach their end, the tutor confirms the winning timeslot and adds a Zoom link to the system. Automated emails are sent to students confirming the group session and requesting them to register for the specific session. Once registered for the session, the students will receive automated emails



with the Zoom link for the session. Students will also receive automated notification emails 24 hours before sessions start. Group session recordings will no longer be uploaded to google drive/DropBox but to the system itself. The system will make use of PayStack to facilitate payments which will allow the system to identify a student that has not made a payment for the upcoming period and disable their access to content and features on the system. Automated emails will be sent out by the system to prompt students to make payments. For individual session bookings, tutors need to add their available timeslots to the system where a student can book one of the available timeslots that suits them best. Once booked, the student and tutor will receive an email from the system informing them of the booked session timeslot.

This proposed system will improve the current processes TDO has and will bring into play new initiatives with the use of technology and the aim of creating a positive experience for all system users.

Conclusion

This section shows the requested systems criteria, which will handle automated communications and eliminate the use of third-party applications. The system will manage session bookings as well as payments, just to mention a few functionalities. It will also make the production of reports more simplistic for the admin users.



4. Preliminary Investigation

Introduction

The Preliminary investigation will provide a synopsis of the goals and objectives that the business seeks to achieve, the challenges/problems the business faces, present opportunities, and directives as well as assumptions and constraints. The elicitation techniques used by the team will also be elaborated on in this section.

4.1 Problem/Vision Statement

	Problem Vision Statement
The problem of	• The current system is extremely fragmented in terms of tuition content delivery and communication. TutorDevOps use a variety of platforms such as Google Drive, DropBox and Discord to teach students. Content needs to be manually distributed to all students from these various platforms. This leads to mismanagement and students will sometimes not receive their appropriate content. Communication is done on a WhatsApp group and tutors are manually required to keep note of all students' individual preferences for sessions which is very tedious for the tutor. Communicating through WhatsApp groups also leads to much miscommunication. • Tracking students' payments is also a manual process. The admin is required to see who has paid/not paid for the upcoming period by cross checking the Wix payment interface and the WhatsApp group. The admin then must manually remove those who have not paid from the group, repos, and email list so that they do not have access to content. • Scheduling group and individual sessions is also a manual process. To schedule a group session, the tutor will post a message on the group with suggestive time slots. The students will then vote for their preferred timeslot and the popular slot is confirmed. This can often lead to decision delays and management errors. The scheduling of individual sessions is done through the student privately messaging the tutor on WhatsApp. The tutor would then have to manually check



	his calendar to confirm a slot. There is often a time delay and confusion when scheduling these sessions.
Affects	The Admin and tutors at TutorDevOps. As well as the students, in terms of their quality of content and service delivery.
The impact of which	The fragmentation of their current system and activities leads to time consuming admin work, mismanagement, and unclear communication.
A successful solution would	A successful solution would create a centralised learning management system that would consolidate all admin activities, content delivery and communication. The system would automate a part of admin's workload. The system would be able to not give access to students who have not paid for the coming period. The system will allow for a simplistic consolidated booking approach, allowing students to vote and register for a session on the system. Further students will be able to access content on the system. Majority of communication is now automated using emails and notifications from the system.

4.2 Business Objectives and Goals

The goals of TutorDevOps are to have central system that performs all the necessary functions to provide an efficient service. The objectives that achieve this goal are the following:

- to consolidate and centralise all admin related activities by November 2021
- to consolidate and centralise all content delivery related activities by November 2021
- to consolidate and centralise all communication related activities by November 2021



4.2.1 Analysis of our Business Goal

We will use the SMART Framework to analyse our above-mentioned business objective.

Framework	Description	Analysis
Specific	The objective is clear and well-defined which in turn allows anyone with basic knowledge of the project to understand the goal.	To consolidate and centralise all admin, content delivery and communication related activities of the system.
Measurable	The objective is measurable in the sense that information concerning the objective can be allocated, detected, or obtained from records.	If all Admin, content delivery and communication related activities are present on one central system.
Achievable	The objective is achievable if, the necessary resources are available and both possible for the business and achieved previously by others in similar circumstances.	It is an achievable objective due to the fact that available technologies and skillsets exist for us to build a central system.
Relevant	The objective is relevant if the team has a clear understanding of how the objective aligns with the overall vision and mission of the business.	A centralised system will allow for the business to be managed more efficiently thus leading to greater growth.
Time-Bound	The objective is timed meaning that there is a clear timeline by which this should be achieved	The objective should be developed and running by November 2021.



4.3 Detailed Problems, Opportunities & Directives Matrix

PROJECT:	LEARN2CODE	PROJECT MANAGER:	Bea Scheepers
CREATED BY:	Vrajesh Patel and Roberto Adinolfi	LAST UPDATED BY:	Vrajesh Patel and Roberto Adinolfi
DATE CREATED:	27/03/2021	DATE LAST UPDATED:	MM/DD/YYYY

Brief Statement of Problem	The impact the problem is causing	Expected benefits from any potential solution	How quickly can the problem potentially be resolved	What is the underlying source of the problem?	What will it cost to solve the problem?
1. Fragmented System in terms of admin and content delivery.	Time consuming admin work and mismanagement.	More effective and efficient admin work with easier access to centrally collated data.	8 months	Various platforms are used to gather data and deliver content to students.	R0; 8 months of hard work from Team 17 of INF370 together with business knowledge of the clients.
2. Communication with regards to admin work, scheduling sessions and student finances is inefficient.	Communication is very manual and tedious which in turn leads to unclear communication and scheduling errors.	Clearer and structured communication due to it being automated. This will lead to easier and accurate scheduling, an increase in students attending lectures and more timeous payments because students will be reminded.	4 months	Use of private and WhatsApp groups to manually communicate with students.	R0; 4 months of hard work from Team 17 of INF370 together with business knowledge of the clients.



Brief Statements of Opportunity	Urgency	Visibility	Annual Benefits	Priority or Rank	Proposed Solution
3. Affective and effective communication through automated emails and in-app notifications.	4 months	Medium	Improve admin efficiency and student attendance	3	Students will be notified through automatic emails and in-app notifications with regards to admin, sessions, and finances.
4. Using a central system to give students access to resources.	4 months	High	Deliver student content more efficiently and accurately.	1	Students will have access to resources on a central system whenever they need.
5. Easier management and tracking of student accounts and finances.	5 months	High	Clearer management of finances and ensuring student payments.	2	Students will be notified when accounts need to be paid and admin will be notified when accounts are in arrears. Should account be in arrears the system will automatically block them from receiving content.

Brief Statements of Directive	Urgency	Visibility	Annual Benefits	Priority or Rank	Proposed Solution
6. The fragmented system will be converted into a central system.	7 months	High	All tasks can be performed on one system thus saving time	1	Creation central system.



4.4 Preliminary Assumptions

Assumptions

- Students need to register on the system.
- Students can book sessions and access content through the system.
- Once students' book for a session, it is assumed that the student will not cancel.
- Tutors need to be registered by admin. Admin will send them a username and password.
- Automatic emails to be sent as communication.
- TutorDevOps has the technological resources to make use of the system.
- Individual tutors will have resources to deliver content.
- Payments will be made before the start of each upcoming period.

Constraints

- Budget: there is no monetary budget available to deliver this project. It will be delivered through the hard work of the Installing Inspiration team.
- Technology: Students will initially have resistance and a learning curve to adopting a new system.
- Skills: Installing Inspiration will need to develop the skills to produce the system.
- Time: The system must be developed by the 18th of October.



4.5 Elicitation Techniques

Elicitation Techniques	Motivation
Interview	Due to Covid-19, the interview was conducted over Google Meets. Our team of 5 were present to interview the 2 owners of TutorDevOps. We made use of the interview technique as it allowed us to identify facts and opinions regarding their current system and business processes as well aspirations for the new system. Meeting each other 'face-to-face' over video call allowed us to develop a relationship with the client and understand the character of the client.
Brainstorming	After interviewing the client our team met on a few occasions to brainstorm a system and its proposed functional requirements. Brainstorming allowed for an open environment where our team were able to provide their opinions and ideas for the new proposed system. The ideas and opinions were then collected and analysed to select the most, creative, and appropriate ideas. This was the ideal way to collect creative open ideas from our team members.
Workshop	After a few brainstorming and analysing sessions our group then travelled to Hatfield, Pretoria to conduct a workshop with the clients in a Covid safe environment. The team presented their proposed system and functional requirements to the clients beforehand. Which were then analysed and deeply discussed in this workshop. Due to some very tough discussions, the workshop allowed us to gain consensus on the requirements of the proposed system and in turn relationships between both team members and clients were strengthened.



4.7 Elicitation Technique Documentation

4.7.1 Interview

Interviews

Preparation checklist

No	Task	Done
1	Decide which type of interview	☒
2	Decide on interview goal	☒
3	Create list of questions	☒
4	Identify potential interviewees	☒
5	Decide on location for interview	☒
6	Invite interviewees	☒
7	Send questions to interviewees (optional)	☐

During Elicitation checklist

No	Task	Done
1	Describe purpose of interview	☒
2	Confirm interviewees' roles	☒
3	Address any concerns	☒
4	Explain how information will be recorded and shared	☒
5	Ask predefined questions	☒
6	Summarize the session	☒

After Elicitation checklist

No	Task	Done
1	Organize information	☒
2	Confirm results with interviewees	☒
3	Share information with stakeholders	☒
4	Schedule follow-up interview if needed	☒

**Preparation:**

1. Due to Covid-19, the interview was conducted over Google Meets. The team opted for an Informal meeting with the owners of TutorDevOps as this allowed for a friendly and relaxed atmosphere.
2. The primary goal of the interview was to gain an understanding of how TutorDevOps current system works and the potential system they want developed. The secondary goal was to gain an understanding of the company's ideology and objectives so that our proposed system could align with those.
3. The following questions along these lines were asked:
 - Provide an explanation of who you are and what role you play at TutorDevOps?
 - What service does TutorDevOps provide and to whom?
 - Describe the organisational structure of TutorDevOps?
 - Describe the Vision and mission for your company?
 - What goals and objectives do you have for the company?
 - Where do you see this business in one year?
 - Describe how the current system of TutorDevOps works?
 - What shortfalls do you see with your current system?
 - What functionality would you like in a newly developed system?
 - What aspects of the current system would you like to see in the new system?
4. The interviewee was Calvin Munava, the co-owner of TutorDevOps.
5. Due to Covid-19, the interview was conducted over Google Meets.
6. Requested an interview over the phone.
7. We did not send the questions to the interviewee prior to the meeting.

During Elicitation:

1. During the beginning stages of the interview, we explained to the purpose of the interview to the interviewee and that was for us to gain an understanding of how TutorDevOps current system works and the potential system they want developed.
2. The role of the interviewee namely Calvin Munava, who is the co-owner was to describe how TutorDevOps operates and answer any questions.
3. The concerns we addressed were:
 - The need for a centralized system by bringing all the platforms together.
 - If the client would have meetings on a regular basis.



- the client would like to change the way they communicate with students by means of email rather than WhatsApp.
- 4. We notified Calvin Munava that the Google Meets session was being recorded and gained his permission, secondly team members took down minutes and notes of the interview.
- 5. We prepared questions prior to the interview.
- 6. The session was summarized by team members in the form of notes.

After Elicitation

1. The team met up after the interview to discuss the outcomes and organize the information we gathered.
2. A write up of the current system was provided to the interviewee to confirm details.
3. The Writeup was share with the owners and all members of the Inspiring Inspiration team.
4. The second meeting has been scheduled on the 1st of April.

4.7.2 Brainstorming

Brainstorming

Preparation checklist		
No	Task	Done
1	Define area of interest	☒
2	Define time limit	☒
3	Identify participants	☒
4	Identify facilitator	☒
5	Invite participants	☒
6	Invite facilitator	☒
7	Meet with participants to explain expectations	☒
8	Establish evaluation criteria	☒
9	Book venue/meeting room	☒

During Elicitation checklist		
No	Task	Done
1	Share new ideas	☒
2	Record all ideas	☒
3	Build on ideas	☒
4	Elicit as many ideas as possible	☒

After Elicitation checklist		
No	Task	Done
1	Discuss and evaluate ideas	☒
2	Create list of ideas	☒
3	Rate ideas	☒
4	Distribute final list of ideas	☒
5	Schedule follow-up if needed	☐

Preparation:

1. The area of interest for the first brainstorming session was to discuss ideas and concerns. The following sessions were concerned with identifying requirements and functional capabilities for our system.
2. The brainstorming sessions were spread out over a week which included two hours every second day.



3. All the members of Installing Inspiration.
4. The group leader namely, Bea Scheepers was the facilitator.
5. All group members were invited.
6. The group leader was invited.
7. The participants of the session were explained what exactly the goals and outcomes of the session were meant to be.
8. The evaluation criteria for idea were clarity and practicality
9. Google meets was used.

During Elicitation

1. All members thought about potential ideas prior to the session and thus they were discussed during the session.
2. The following ideas were identified:
 - Bookings and scheduling of sessions to be done on the system
 - Business reports are essential
 - Payments to be done on system
 - Communication should be automated through the use of emails and notifications
 - Packages should be sold on the system
 - Resources such as past videos should be accessible on the system
 - People should be able apply to be a Tutor
 - Students shouldn't be able to download and share resources
 - Tutors and students should have accounts
 - Login functionality needed
3. We took the ideas from various platforms that TutorDevOps use and built on them to come up with our own ideas and solutions.
4. The session was open to all ideas and this allowed for an influx of ideas from every team member.

After Elicitation

1. The ideas provided above were discussed in detail among team members to filter them out and identify the best and most realistic ideas.
2. The list of ideas is the following:
 - Bookings and scheduling of sessions to be done on the system
 - Business reports are essential



- Payments to be done on system
 - Communication should be automated using emails and notifications
 - Packages should be sold on the system
 - Resources such as past videos should be accessible on the system
 - People should be able apply to be a Tutor
 - Tutors and students should have accounts
 - Login functionality needed
3. We rated our ideas based on complexity and practicality.
 4. The ideas and notes of the session were uploaded on a Google Drive.

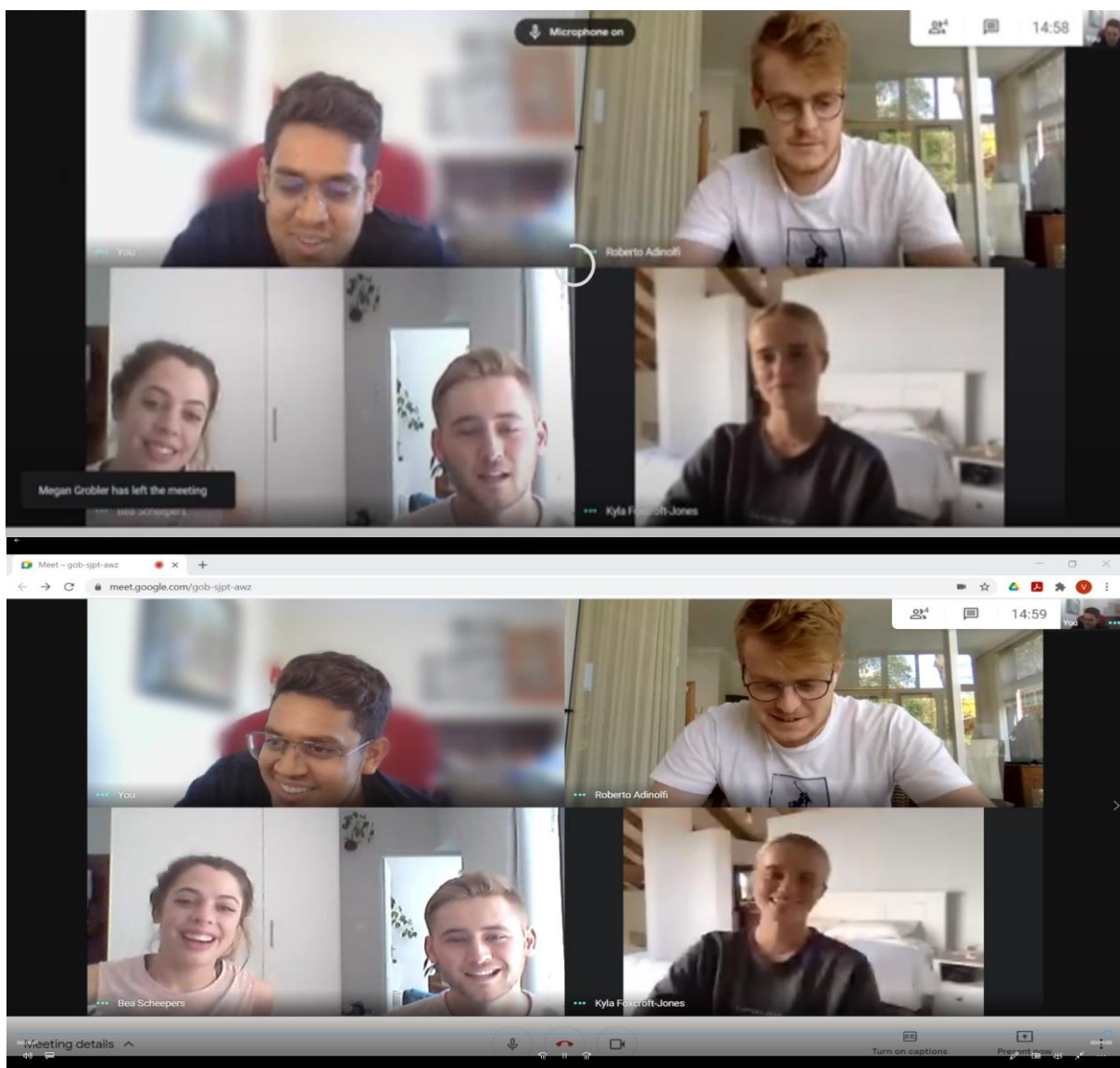


Figure 4 - Screenshots of Brainstorm Sessions



4.7.3 Workshop

Workshops

Preparation checklist		
No	Task	Done
1	Define purpose and desired outcomes	☒
2	Identify stakeholders	☒
3	Identify facilitator and scribe	☒
4	Create agenda	☒
5	Determine how outputs will be captured	☒
6	Arrange venue/meeting room	☒
7	Arrange equipment	☒
8	Send agenda	☒
9	Conduct pre-workshop interviews	☒
10	Define workshop roles	☒

During Elicitation checklist		
No	Task	Done
1	State purpose and desired outcomes	☒
2	Establish agreed-upon ground rules	☒
3	Work through agenda	☒

After Elicitation checklist		
No	Task	Done
1	Document results	☒
2	Follow-up on open action items	☒
3	Distribute documentation to attendees and stakeholders	☒

Preparation:

1. The purpose and desired outcome of the workshop was to develop stronger relationship with the clients and gain a deeper understanding of the company's background and future need.
2. The Stakeholders of the workshop would our team and the owners of TutorDevOps.
3. Our group leader Bea Scheepers was the facilitator and Stephan Bergh was the scribe.



4. The Agenda was created and sent to the clients and is provided in section 4.8.
5. Outputs were captured in the form of taking notes.
6. The workshop took place at the residence of one of our team members in Hatfield, Pretoria.
7. Attendees of the workshop brought their own equipment such as laptops.
8. The agenda was sent to all attendees prior to the workshop.
9. Interviews were conducted as described in the above elicitation technique.

10. Our group leader

Bea Scheepers
was the
facilitator,
Stephan Bergh,
the rest of the
team was tasked
with taking notes.



Figure 5 - Mural of Workshop Sessions

During Elicitation:

1. The purpose and desired outcomes that were mentioned above were stated and described to all attendees during the workshop.
2. Basic ground rules were established such as:
 - Be respectful
 - Stay focused both mentally
 - Contribute to the discussions
 - Openly listen to attendees' contributions
 - Be on time and respect the process
3. The agenda was worked through step by step.

After Elicitation:

1. The outcome and results of the workshop were compiled in a document.
2. Open action items were handed to team members as responsibilities to complete for our next scheduled meeting.
3. All documentation compiled after the workshop was uploaded to the google drive for team members to access and provided to the owners of TutorDevOps.



4.8 Details of Elicitation Tasks

4.8.1 Interview Details

Prior to the interview, the Installing Inspiration team sat down together and formulated possible questions for the potential client. The questions as seen below were formulated to gain an understanding of how TutorDevOps operates and what functionality they would need in a new system to improve their way of operating.

During the interview, the questions below were asked and in return the answers from Calvin Munava who a co- founder of TutorDevOps were recorded as evident below.

- **Provide an explanation of who you are and what role you play at TutorDevOps?**

Calvin Munava is a co-founder of TutorDevOps, he works as a software developer and in terms of TutorDevOps he holds the position of tutor and admin., as a co-founder Calvin helps steer the company and provides insight on how the company should operate.

- **What service does TutorDevOps provide and to whom?**

TutorDevOps provides a teaching/tutoring service to students of various institutions that need help with programming and software development. The service includes both group sessions and private sessions which are hosted online, Resources are provided to students such as homework assignments, recorded videos, and notes/cheat sheets. The company also offers its services to the general public.

- **Describe the organisational structure of TutorDevOps?**

The company consists of the two co-founders namely, Calvin Munava and Iain Steyn, an accountant by the name of Lillian, an Admin assistant by the name of Palesa and three tutors excluding the co-founders namely, Marco, Cody and Eppy. Thus, the structure of the business will be that of a *Simple Structure* as both co-founders handle all major decisions and oversee all operations with help of a small staff.

- **Describe the Vision and mission for your company?**

The vision and mission are to be able to tutor and help students fulfil their potential when it comes to programming in a manner that is inclusive to all individuals in South Africa regardless of a language barrier.

- **What goals and objectives do you have for the company?**

By the end of the year 2021 TutorDevOps aims to have a strong foothold in South Africa's top universities to provide a service to most higher institute students. TutorDevOps also aims to



provide a tutoring service to high school students and both these goals are achievable through a central system.

- **Where do you see this business in one year?**

The business will be able to provide a service to both university and high school students and thus having a larger customer base.

- **Describe how the current system of TutorDevOps works?**

The current system is spread across multiple platforms, a home page is hosted on Wix which consists of information on the company and the functionality to buy packages (Noob and Pro), Group sessions are hosted twice a week either on zoom or discord, all resources based on the session are then provided to the student via DropBox. Communication is done using group WhatsApp messages.

- **What shortfalls do you see with your current system?**

The shortfall with the current system is that students can download resources and share then with individuals that are not customers of TutorDevOps, communication and admin work is a manual task and booking of sessions is tedious as tutors must manually check the WhatsApp group to see what time suits everyone. There is no portal for students to login and access resources. Providing resources through Dropbox is tedious as a tutor must sift through all the emails of the students to share resources to them.

- **What functionality would you like in a newly developed system?**

On a central system there should be a portal that allows student and tutors to login and access their accounts. Tutors should be able to schedule timeslots for sessions on the system and thus allowing students to view the timeslots in a calendar format and vote on which session best suits their schedule, individual sessions should be booked through the system and this can be done through the calendar. Communication regarding payments and sessions should be automated using emails and in app messages. Packages and courses can be purchased through the system and all resources based on sessions will be stored on a database and accessed through the system.

- **What aspects of the current system would you like to see in the new system?**

The potential system should have the same packages and the home page on Wix should be the same.

After the interview, the Installing Inspiration team got together and discussed the feedback/ answers we received from the interview and concluded that we would take them on as a client and that a brainstorming session will be needed to formulate ideas.

4.8.2 Brainstorming Details

It was decided amongst the Installing Inspiration team that after the first interview with TutorDevOps we should have a brainstorming session. The aim of these sessions was to come up with ideas and possible functional requirements for our potential system. The ideas and opinions were then collected and analysed, and the most creative and appropriate ideas were selected and put in a mind map to help visualize the results of the session. Attached below is a mind map that holds the ideas that came out of the session.



Figure 6 - Mind map from Brainstorm session



4.8.3 Workshop Details

The team approached the owners of TutorDevOps to set up a potential workshop we believed it would help us to deepen our understanding of the current system and the goals and objectives of the company. The team presented their proposed system and functional requirements to the clients prior to the workshop and in turn they were analysed and deeply discussed in this workshop. Due to some very tough discussions, the workshop allowed us to gain consensus on the requirements of the proposed system and in turn relationships between both team members and clients were strengthened.



Figure 7 - Mural of Workshop Session

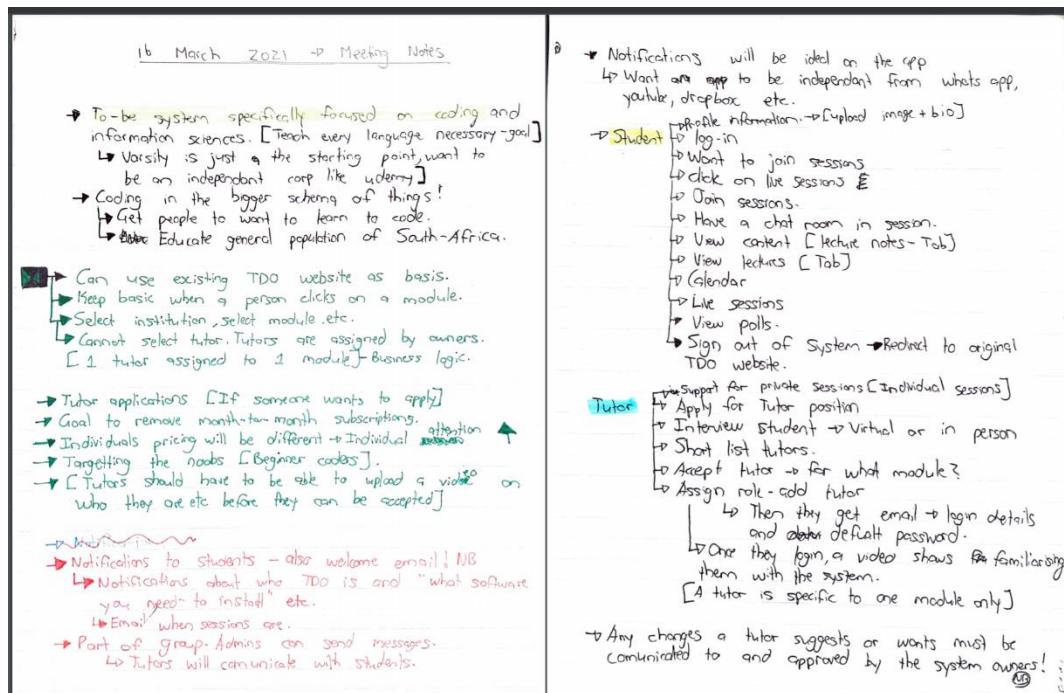


Figure 8 - Scan of notes from Workshop Session

Conclusion

The current system used by TutorDevOps is flawed to a large extent. The many flaws can be attributed to the fragmentation of the system and lack of cohesion between the differing platforms. Thus, it is evident that the newly proposed system can resolve/ mitigate these flaws as it has the functional requirements needed for TutorDevOps to achieve their objectives and goals.



5. Problem Analysis

Introduction

This section provides a description of the current system being used by TutorDevOps in the form of a rich picture as well as an analysis of the current system's capabilities, the required capabilities, capability gaps and recommendations to resolve these gaps in tabular format.

5.1 Current System Rich Picture

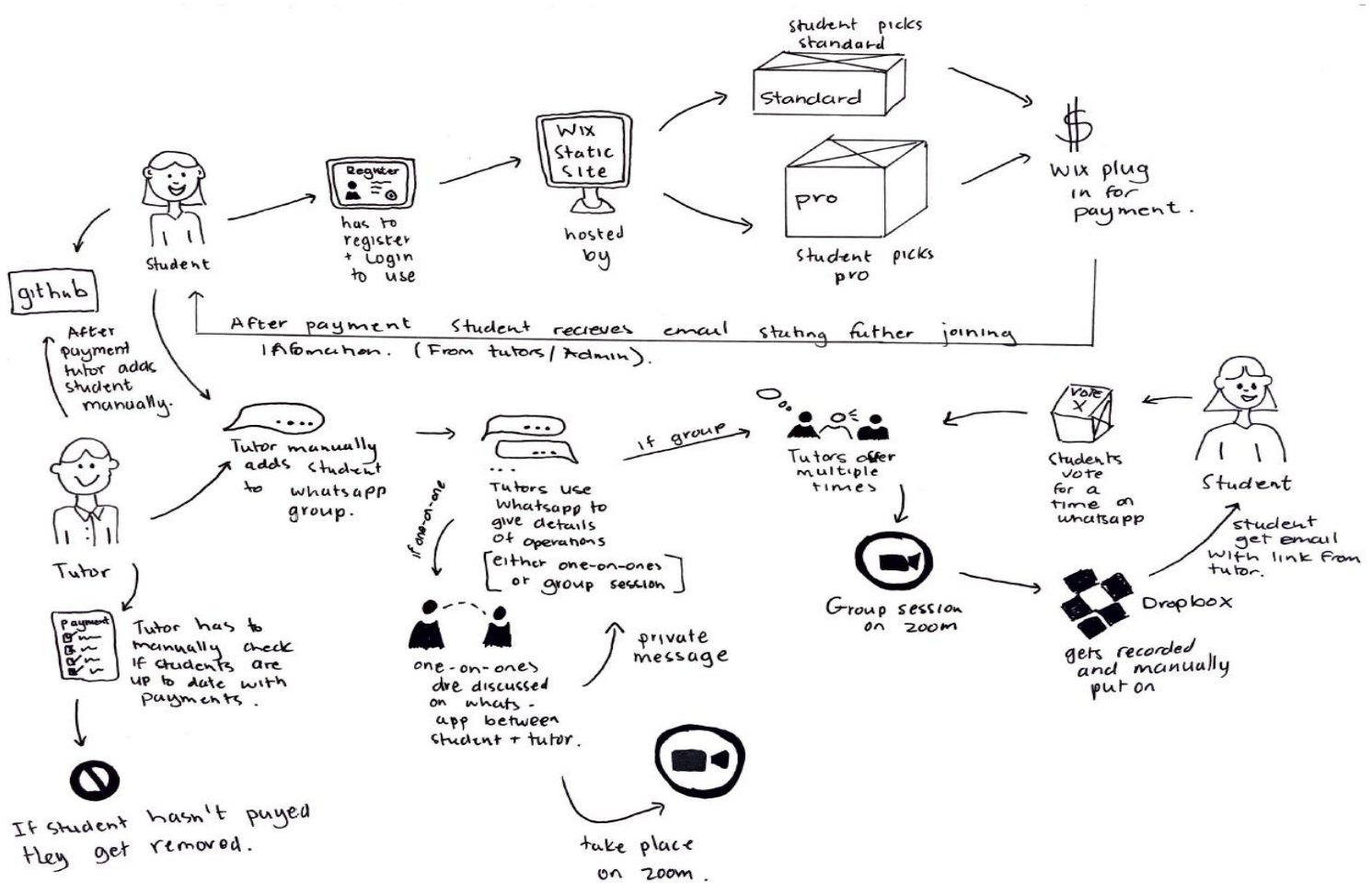


Figure 9- Picture of Current System Rich Picture



5.2 Current System Capability and Gap Analysis

	Current System Capabilities	Proposed System Capabilities	Capability Gaps	Plan to solve capability Gap
	<i>The current state of the system</i>	<i>The aim of the final state of the system</i>	<i>Gaps not allowing current system to function like final system</i>	<i>Recommendations of actions needed to fill gaps</i>
1	A student registers on their Wix Static Site with their relevant information. After which they proceed to purchase a tuition group or individual package. On completion of this purchase the student receives an email stating further joining information. The student is then manually added to the WhatsApp groups and GitHub repos.	A student registers on the online system and their data is saved in a secured location. The student is then able to purchase a tuition group or individual package. Upon confirmation of this purchase, the student then automatically receives a welcome email with further information regarding TutorDevOps and their tutors. The student will receive information regarding how to attend group sessions, further book individual sessions and communication channels.	1. Communication is manually generated by the admin across various WhatsApp groups.	1. To remove WhatsApp and to ensure communication through automatic emails and in-app notifications.
2	TutorDevOps offer groups sessions to all registered students for that modules twice a	To schedule a group session, the tutor will create a voting poll on the system giving students three available	1. Scheduling of sessions in manually agreed upon on a WhatsApp group.	1. Creating of a voting mechanism for students to vote for their



	week. The scheduling of these sessions is done through voting on WhatsApp groups. A tutor will send out a message with a different time for students to vote. Each student will reply with their vote. The tutor will then manually tally the votes and send out a second message with confirms the most voted the time as the session time. The session is now scheduled, and the tutor will manually send out a Zoom link on WhatsApp.	slots to choose from. A student is then able to vote for their preferred timeslot. Once the end time has reached for the poll, the tutor is notified of the most popular timeslot. The tutor confirms the time slot and adds a Zoom link to the system. An automatic email is sent out to students informing them of the confirmed group session time. Students are then required to register for session on the system. Once the student is registered an automatic email is sent to the student with the session zoom link and any other extra information. Registration will close 15 minutes before the start of the session.	2. Tutor is required to manually tally student votes and announce the popular session. 3. Tutor is required to manually send zoom link.	preferred session on the system. 2. System will need to inform tutor of the most popular slot. Tutor will need to confirm the slot an automatic email will be sent to all students to register. 3. Once registered, students receive an email with the Zoom link.
3	The tutor will manually send out reminders to the WhatsApp groups to remind students of the	All registered students will receive an email 24 hours before the session reminding them to attend the group session. Similarly, should	1. No options currently exist on the system to send emails to students regarding reminder,	1. To automatically send email to students regarding reminder,



	session. The same is done should there be a cancellation or re-scheduling of a session.	the event be postponed, or cancelled students will receive an email noting the reason for the change and a new timeslot.	postponements, or cancellations.	postponements, or cancellations.
4	Group session recordings are manually uploaded and sent to students via email with DropBox.	Group session recordings are uploaded to the system and all students registered for that module will have access to the recording on the system 48 hours after the session.	1. Sessions are uploaded to DropBox and manually emailed to students.	1. Recordings are uploaded to the system to allow students to access the recording from a centralised point.
5	When the payment schedule comes to an end the admin, through the Wix payment gateway and eft, can note who has not paid for the coming period. Those students are then manually removed from the WhatsApp group and GitHub Repos.	The system will make use of PayStack to facilitate payments. The system will identify that a student has not paid for the upcoming period and will automatically disable the student from accessing content and features on the system. An automatic email will be sent to the student requesting them to pay for the upcoming period.	1. Should a student not be up to date with payments, they must be manually removed from receiving access to content through WhatsApp and GitHub.	1. For the system to identify those that have not paid and to disable them from accessing content and features till payment is received.
6	Should a student choose an individual	Tutors will be required to list their available time slots with zoom	1. Students cannot book a slot on a centralised table	1. Tutor will need to provide their available



	package, they will need to privately message the tutor on WhatsApp to check each other's schedules and book an individual lesson.	links on the system for the upcoming week. Should a student choose an individual package, the student will be able to view available timeslots on the system. The student will then be able to book their preferred available timeslot. Once booked, the student and tutor will receive an email informing them of the confirmed booked timeslot.	and need to privately message the tutor to agree on a timeslot. The student is required to manually communicate with and collate all students to ensure he/she has no clashes.	timeslots with zoom links for the upcoming week. Students will then be able to book available timeslots that they prefer on a centralised system. Once booked, student and tutor will receive an email informing them of the booked the timeslot.
--	--	--	---	--



5.3 Proposed System Rich Picture

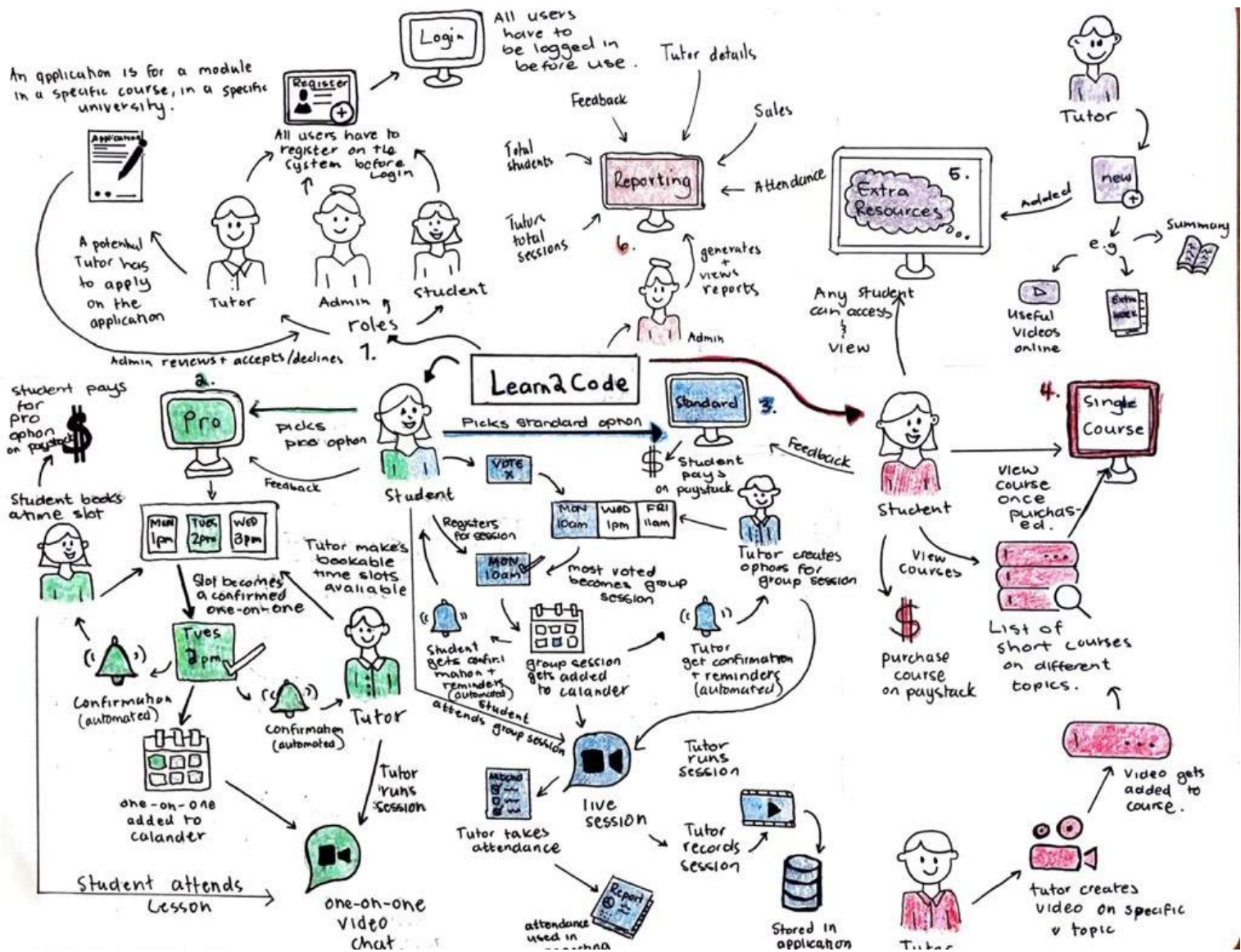


Figure 10 - Picture of Proposed Rich Picture



5.3.1 Table for explaining proposed rich picture

1. Administration	This section illustrates the different user roles (admin, tutor, and student) and they all must be registered and logged in before they can use the system. Potential tutors must apply to be a tutor through the system, and they are accepted or denied by admin (they do not need to be registered/ logged in to apply). A tutor can only apply to be a tutor for a module in a specific course at a specific university.
2. Pro package	This section illustrates: • The booking of a one-on-one session and how it is added to the system calendar. • The automatic confirmation and reminder notifications for both student and tutor. • How the one-on-one sessions take place. • The communication that takes place on the system between a tutor and a single student. • Payment for the pro package on the system (using PayStack).
3. Standard package	This section illustrates: • Communication that takes place on the system between a single tutor and a group of students for confirming a session time and date. • Voting and registration by the students for the sessions placed on the system calendar. • Automatic confirmation and reminder notifications for both students and tutor. • How the one-on-one sessions take place on a video chat. • How the sessions are recorded and stored on the system for students to refer back to. • How attendance is taken by the tutors for reporting purposes. • Payment for the pro package (using PayStack)
4. Single courses	This section illustrates: • Student can purchase a single course (containing small videos) on a specific topic on the system. • Payment on the system for the single course using PayStack. • How a student views the course on the system. How a tutor creates the videos and adds them to a specific course.
5. Resources	This section illustrates where a tutor can add extra resources for students to access in their own time. This may be external videos, homework, or summaries.
6. Reporting	This section illustrates the different reports that are generated by the system and viewed by the admin or tutors for improved business decisions and insight. • Tutor details report. • Sales report. • Attendance report • Feedback report. • Total students report. • Tutors' total session report.



Conclusion

The current system is extremely fragmented across a variety of platforms and thus it must be converted into singular/central system which in turn provides all the functional capabilities and requirements needed to deliver an efficient tutoring service to their customers interface and database to reduce effort and protect information. By centralizing the system admin work will be reduced and in turn human error will also be reduced.



6. Requirement Analysis

Introduction

The following section differentiates between non-functional and functional requirements. Functional requirements are also split up into different sub-systems, 6 sub-systems in total. User acceptance criteria is included per functional requirement.

6.1 Functional requirement list

1. Admin Subsystem

- 1.1 Search tutor application
- 1.2 Maintain tutor application
- 1.3 Create Tutor
- 1.4 Search Tutor
- 1.5 Maintain Tutor
- 1.6 Create university
- 1.7 Search university
- 1.8 Maintain university
- 1.9 Create degrees
- 1.10 Search degrees
- 1.11 Maintain degrees
- 1.12 Create Module
- 1.13 Search Module
- 1.14 Maintain Module
- 1.15 Create Course
- 1.16 Search Course
- 1.17 Maintain Course
- 1.18 Create Subscription



- 1.19 Search Subscription
- 1.20 Maintain Subscription
- 1.21 Search Student
- 1.22 Maintain Student
- 1.23 Create session content category
- 1.24 Search session content category
- 1.25 Maintain session content category
- 1.26 Create course content category
- 1.27 Search course content category
- 1.28 Maintain course content category
- 1.29 Add course content
- 1.30 Search course content
- 1.31 Maintain course content

2. Login Subsystem

- 2.1 Login
- 2.2 Logout
- 2.3 Reset Password
- 2.4 Change Password
- 2.5 Add user role
- 2.6 Search user role
- 2.7 Maintain user role



3. Student Subsystem

- 3.1 Register Student
- 3.2 Maintain Student
- 3.3 Vote for group session
- 3.4 Register for group session
- 3.5 Make individual booking
- 3.6 Search individual booking
- 3.7 Maintain individual booking
- 3.8 Create Message
- 3.9 Search Message
- 3.10 Maintain Message
- 3.11 Add to basket
- 3.12 Remove item/(s) from basket
- 3.13 View basket
- 3.14 Checkout
- 3.15 Create Feedback
- 3.16 Search Feedback
- 3.17 Maintain Feedback

4. Tutor Subsystem

- 4.1 Apply to become a tutor
- 4.2 Create group session vote
- 4.3 Search group session vote
- 4.4 Maintain group session vote
- 4.5 Tutor confirms group voted session



- 4.6 Create group session registration
- 4.7 Search group session registration
- 4.8 Maintain group session registration
- 4.9 Create session
- 4.10 Search session
- 4.11 Maintain session
- 4.12 Take attendance
- 4.13 Search Attendance
- 4.14 Maintain Attendance
- 4.15 Add group session content
- 4.16 Search group session content
- 4.17 Maintain group session content
- 4.18 Search Student
- 4.19 Create individual timeslots
- 4.20 Search timeslot
- 4.21 Maintain timeslot
- 4.22 Maintain Tutor
- 4.23 Create Resource categories
- 4.24 Search Resource categories
- 4.25 Maintain Resource categories
- 4.27 Create Resource
- 4.28 Search Resource
- 4.29 Maintain Resource
- 4.30 Create Message
- 4.31 Search Message



4.32 Maintain Message

5. Payment Subsystem

5.1 Search Payment

5.2 Generate Invoice

5.3 Generate Payment log

6. Report Subsystem

6.1 Generate Total Student report

6.2 Generate Sales Report

6.3 Generate Payment Report

6.4 Generate Feedback Report

6.5 Generate Attendance Report



6.2 Functional Requirement Description and Details

6.2.1 Admin Sub-system

Functional Requirement	Explanation
Requirement number:	1.1
Requirement name (use case name):	Search Tutor Application
Requirement short description:	The system must allow admin to search a tutor's application on the proposed Learn2Code system.
Requirement detailed description and constraints:	The admin must be able to navigate to a applications tab on the proposed Learn2Code system. The admin user should then be able to enter the applicants first name in a search bar. The system will then return a list of all the applications that match the first name entered in the search bars. Constraint(s) • The admin may provide faulty search criteria which will produce incorrect results.
Business rules applicable to this requirement	• The application must already exist on the system. • The admin must already exist and be logged into the system. • Only the admin user role can search for a tutor application.
Revision date and Revision number:	2021-04-03 Version 1.0
Criticality/Priority:	Low

Functional Requirement	Explanation
Requirement number:	1.2
Requirement name (use case name):	Maintain Tutor Application
Requirement short description:	The system must allow for the admin user to delete a tutors application from the proposed Learn2Code system.
Requirement detailed description and constraints:	The admin must be able to navigate to a tutors application (invoke 1.1 search tutor application) on the proposed Learn2Code system. The admin user should then be able to choose to deny the application. The system will then ask admin for confirmation to delete the tutor



	application. Once confirmed the list of tutor applications will be updated with the removal of the specific tutor application. The applicant will get a automatic notification of their rejection. Constraint(s): Once denied and deleted the admin will not be able to recall the application. If the admin user makes a mistake the potential tutor will have to reapply.
Business rules applicable to this requirement	• Only the admin user role can maintain a tutor application. • Admin must be logged in.
Revision date and Revision number:	2021-04-03 Version 1.0
Criticality/Priority:	Medium

Functional Requirement	Explanation
Requirement number:	1.3
Requirement name (use case name):	Create Tutor
Requirement short description:	The admin should be able to choose to accept and register a new tutor from the applications. The admin then completes the forms with the relevant information and adds the new tutor to the system.
Requirement detailed description and constraints:	The admin navigates the system and locates the tutor application in question. The admin should then be able accept the application and to choose to create a tutor. Admin will then need to complete the on-screen information about the tutor. Admin will have to provide a temporary username and password in the input fields provided. After admin has completed the information they will confirm and add the new tutor to the system. The temporary profile details are automatically sent to the new tutor.
Business rules applicable to this requirement	• Only admin can accept a tutors application. • The application must already exist on the system. • The admin must already exist and be logged into the system.



Revision date and Revision number:	2021-04-03 Version 1.0
Criticality/Priority:	High

Functional Requirement	Explanation
Requirement number:	1.4
Requirement name (use case name):	Search Tutor
Requirement short description:	The admin should be able to search a registered tutor on the proposed Learn2Code system.
Requirement detailed description and constraints:	The admin must be able to navigate to the tutors tab on the Learn2Code system. The admin then should be able to enter the details of the tutor into the search bar on the Lean2Code system. System retrieves a list of tutors (done in real time) that match the tutor details entered into the search bar. Constraint(s): • The admin may provide faulty search criteria which will produce incorrect results.
Business rules applicable to this requirement	• Only tutors that are registered should be displayed, no potential tutors.
Revision date and Revision number:	2021-04-03 Version 1.0
Criticality/Priority:	Low

Functional Requirement	Explanation
Requirement number:	1.5
Requirement name (use case name):	Maintain Tutor
Requirement short description:	The system must allow for the admin user to update or delete a tutor from the proposed Learn2Code system.



Requirement detailed description and constraints:	The admin must be able to navigate to a tutor that is registered (invoke 1.4 search tutor) on the proposed Learn2Code system. The admin user should then be able to choose to maintain the tutor. The system will then display the tutors profile details and an option to delete the tutor profile. The admin will be able to enter an updated version of those details or be able to delete the tutor and their details from the system. If the updated details are valid or the delete option is chosen, the system will ask for confirmation and complete the maintain request. Constraint(s): None
Business rules applicable to this requirement	• The admin must enter details that still correspond with the required format. • The tutor must already exist on the system for it to be maintained. • The admin must be logged in.
Revision date and Revision number:	2021-04-03 Version 1.0
Criticality/Priority:	Medium

Functional Requirement	Explanation
Requirement number:	1.6
Requirement name (use case name):	Create University
Requirement short description:	The admin should be able to create a university category which courses can then be created under.
Requirement detailed description and constraints:	The admin must be able to navigate to the content tab on the Learn2Code system. The admin should then be able to choose to create/add a new university category. They will have to give the university name in the input field provided. After admin has completed the information they will confirm and add the new university to the system. Constraint(s)



	none
Business rules applicable to this requirement	none
Revision date and Revision number:	2021-04-03 Version 1.0
Criticality/Priority:	High

Functional Requirement	Explanation
Requirement number:	1.7
Requirement name (use case name):	Search University
Requirement short description:	The admin should be able to search a university on the proposed Learn2Code system.
Requirement detailed description and constraints:	The admin must be able to navigate to the university section on the content tab on the Learn2Code system. The admin then should be able to enter the name of the university into the search bar on the Lean2Code system. System retrieves the university and any subcategories under it (done in real time) that match the name entered into the search bar. Constraint(s): • The admin may provide faulty search criteria which will produce incorrect results.
Business rules applicable to this requirement	• The university has to be created before it can be searched.
Revision date and Revision number:	2021-04-03 Version 1.0
Criticality/Priority:	Low

Functional Requirement	Explanation
Requirement number:	1.8



Requirement name (use case name):	Maintain University
Requirement short description:	The system must allow for the admin user to update or delete a university category from the proposed Learn2Code system.
Requirement detailed description and constraints:	The admin must be able to navigate to a university category (invoke 1.7 search university) on the proposed Learn2Code system. The admin user should then be able to choose to maintain the university. The system will then display the university name and the information under it, also an option to delete the university. The admin will be able to enter an updated version of university name or be able to delete the university and its details from the system. If the updated name is valid or the delete option is chosen, the system will ask for confirmation and complete the maintain request. Constraint(s): Once university is deleted all the degree and subsequently the modules within the degrees are also deleted. These cannot be retrieved and will have to be readded.
Business rules applicable to this requirement	• The admin must enter details that still correspond with the required format. • The university must already exist on the system for it to be maintained. • The admin must be logged in.
Revision date and Revision number:	2021-04-03 Version 1.0
Criticality/Priority:	Medium

Functional Requirement	Explanation
Requirement number:	1.9
Requirement name (use case name):	Create Degrees
Requirement short description:	The admin should be able to create a course under a university category which modules can then be created under.
Requirement detailed description and constraints:	The admin must be able to navigate to the universities on the Learn2Code system. The admin should then be able to choose to create/add a new degree category within a specific university. They will have to give the degree name in the input field provided. After admin



	has completed the information, they will confirm and add the new degree to the system. Constraint(s) none
Business rules applicable to this requirement	A university has to first exist before a degree can be added to it.
Revision date and Revision number:	2021-04-03 Version 1.0
Criticality/Priority:	High

Functional Requirement	Explanation
Requirement number:	1.10
Requirement name (use case name):	Search Degrees
Requirement short description:	The admin should be able to search a degree on the proposed Learn2Code system.
Requirement detailed description and constraints:	The admin must be able to navigate to the degrees under the university tab on the Learn2Code system. The admin then should be able to enter the name of the degree into the search bar on the Lean2Code system. System retrieves the degree and any subcategories and information under it (done in real time) that match the name entered into the search bar. Constraint(s): The admin may provide faulty search criteria which will produce incorrect results
Business rules applicable to this requirement	The degree has to be created before it is able to be searched.
Revision date and Revision number:	2021-04-03 Version 1.0
Criticality/Priority:	Low



Functional Requirement	Explanation
Requirement number:	1.11
Requirement name (use case name):	Maintain Degrees
Requirement short description:	The system must allow for the admin user to update or delete a degree category from the proposed Learn2Code system.
Requirement detailed description and constraints:	The admin must be able to navigate to a degree category (invoke 1.10 Search Degrees) on the proposed Learn2Code system. The admin user should then be able to choose to maintain the degree. The system will then display the degree name and the information under it, also an option to delete the degree. The admin will be able to enter an updated version of degree name or be able to delete the degree and its details from the system. If the updated name is valid or the delete option is chosen, the system will ask for confirmation and complete the maintain request. Constraint(s): Once a degree is deleted all the modules within the degrees are also deleted. These cannot be retrieved and will have to be readded.
Business rules applicable to this requirement	• The admin must enter details that still correspond with the required format. • The degree must already exist on the system for it to be maintained. • The admin must be logged in.
Revision date and Revision number:	2021-04-03 Version 1.0
Criticality/Priority:	Medium

Functional Requirement	Explanation
Requirement number:	1.12
Requirement name (use case name):	Create Module
Requirement short description:	The admin should be able to create a module category within a degree category which content can then be added to.



Requirement detailed description and constraints:	The admin must be able to navigate to the degrees under the university category on the Learn2Code system. The admin should then be able to choose to create/add a new module category within a specific degree. They will have to give the module name in the input field provided. After admin has completed the information they will confirm and add the new module to the system. Constraint(s) none
Business rules applicable to this requirement	The degree in which the module belongs has to exist first before a module can be added.
Revision date and Revision number:	2021-04-03 Version 1.0
Criticality/Priority:	High

Functional Requirement	Explanation
Requirement number:	1.13
Requirement name (use case name):	Search Module
Requirement short description:	The admin should be able to search a module on the proposed Learn2Code system.
Requirement detailed description and constraints:	The admin must be able to navigate to the modules in a degree under the university tab on the Learn2Code system. The admin then should be able to enter the name of the module into the search bar on the Lean2Code system. System retrieves the module and information under it (done in real time) that match the module name entered into the search bar. Constraint(s): The admin may provide faulty search criteria which will produce incorrect results
Business rules applicable to this requirement	The module has to be created before it is able to be searched.



Revision date and Revision number:	2021-04-03 Version 1.0
Criticality/Priority:	Low

Functional Requirement	Explanation
Requirement number:	1.14
Requirement name (use case name):	Maintain Module
Requirement short description:	The system must allow for the admin user to update or delete a module category from the proposed Learn2Code system.
Requirement detailed description and constraints:	The admin must be able to navigate to a module category (invoke 1.13 Search Module) on the proposed Learn2Code system. The admin user should then be able to choose to maintain the module. The system will then display the module name and the information under it, also an option to delete the module. The admin will be able to enter an updated version of module name or be able to delete the module and its details from the system. If the updated name is valid or the delete option is chosen, the system will ask for confirmation and complete the maintain request. Constraint(s): Once a module is deleted all the data within the module is also deleted. These cannot be retrieved and will have to be readded.
Business rules applicable to this requirement	• The admin must enter details that still correspond with the required format. • The degree must already exist on the system for it to be maintained. • The admin must be logged in.
Revision date and Revision number:	2021-04-03 Version 1.0
Criticality/Priority:	Medium

Functional Requirement	Explanation
Requirement number:	1.15



Requirement name (use case name):	Create Course
Requirement short description:	The admin should be able to create a single course under the courses tab on Learn2Code nav bar.
Requirement detailed description and constraints:	The admin must be able to navigate to the courses on the nav bar of the Learn2Code system. The admin should then be able to choose to create/add a new course. They will have to give the course name in the input field provided. After admin has completed the information they will confirm and add the new course to the system. Constraint(s) none
Business rules applicable to this requirement	Only the admin user role can create a course within the system
Revision date and Revision number:	2021-04-03 Version 1.0
Criticality/Priority:	High

Functional Requirement	Explanation
Requirement number:	1.16
Requirement name (use case name):	Search Course
Requirement short description:	The admin should be able to search a single course on the proposed Learn2Code system.
Requirement detailed description and constraints:	The admin must be able to navigate to the courses tab on the Learn2Code system navigation bar. The admin then should be able to enter the name of the course into the search bar on the Lean2Code system. System retrieves the course and any information under it (done in real time) that match the course name entered into the search bar. Constraint(s):



	The admin may provide faulty search criteria which will produce incorrect results
Business rules applicable to this requirement	The course has to exist on the system before it can be searched.
Revision date and Revision number:	2021-04-03 Version 1.0
Criticality/Priority:	Low

Functional Requirement	Explanation
Requirement number:	1.17
Requirement name (use case name):	Maintain Course
Requirement short description:	The system must allow for the admin user to update or delete a single course from the proposed Learn2Code system.
Requirement detailed description and constraints:	The admin must be able to navigate to a single course (invoke 1.16 Search Course) on the proposed Learn2Code system. The admin user should then be able to choose to maintain the course. The system will then display the course name and the information under it, also an option to delete the course. The admin will be able to enter an updated version of course name or be able to delete the course and its details from the system. If the updated name is valid or the delete option is chosen, the system will ask for confirmation and complete the maintain request. Constraint(s): Once a course is deleted all the data added by the tutor within the course is also deleted. These cannot be retrieved and will have to be readded.
Business rules applicable to this requirement	• The admin must enter details that still correspond with the required format. • The course must already exist on the system for it to be maintained. • The admin must be logged in.
Revision date and Revision number:	2021-04-03 Version 1.0
Criticality/Priority:	Medium



Functional Requirement	Explanation
Requirement number:	1.18
Requirement name (use case name):	Create Subscription
Requirement short description:	The admin should be able to create a subscription for the students to sign up to.
Requirement detailed description and constraints:	The admin must be able to navigate to the subscription on the nav bar of the Learn2Code system. The admin should then be able to choose to create/add subscription. They will have to give the subscription name in the input field provided. After admin has completed the information they will confirm and add the new subscription to the system.
Business rules applicable to this requirement	Only the admin user role can create a subscription within the system
Revision date and Revision number:	2021-04-07 Version 1.0
Criticality/Priority:	High

Functional Requirement	Explanation
Requirement number:	1.19
Requirement name (use case name):	Search subscription
Requirement short description:	The admin should be able to search a single subscription on the proposed Learn2Code system.
Requirement detailed description and constraints:	The admin must be able to navigate to the subscription tab on the Learn2Code system navigation bar. The admin then should be able to enter the name of the subscription into the search bar on the Learn2Code system. System retrieves the subscription and any information under it (done in real time) that match the subscription entered into the search bar. Constraint(s): The admin may provide faulty search criteria which will produce incorrect results



Business rules applicable to this requirement	The subscription has to exist on the system before it can be searched.
Revision date and Revision number:	2021-04-07 Version 1.0
Criticality/Priority:	High

Functional Requirement	Explanation
Requirement number:	1.20
Requirement name (use case name):	Maintain subscription
Requirement short description:	The system must allow for the admin user to update or delete a single subscription from the proposed Learn2Code system.
Requirement detailed description and constraints:	The admin must be able to navigate to a single course (invoke 1.19 Search subscription) on the proposed Learn2Code system. The admin user should then be able to choose to maintain the subscription. The system will then display the subscription name and the information under it, also an option to delete the subscription. The admin will be able to enter an updated version of subscription name or be able to delete the subscription and its details from the system. If the updated name is valid or the delete option is chosen, the system will ask for confirmation and complete the maintain request. Constraint(s): Once a subscription is deleted all the data added by the tutor within the subscription is also deleted. These cannot be retrieved and will have to be readded.
Business rules applicable to this requirement	• The admin must enter details that still correspond with the required format. • The subscription must already exist on the system for it to be maintained. • The admin must be logged in.
Revision date and Revision number:	2021-04-07 Version 1.0
Criticality/Priority:	High



Functional Requirement	Explanation
Requirement number:	1.21
Requirement name (use case name):	Search Student
Requirement short description:	The system will allow the admin through its interface to search for all existing students and their related accounts and details
Requirement detailed description and constraints:	The system allows for student's accounts created by the student to be searched by the admin. The system will search for the student by using specific search criteria like first name or first letters of surname. The system will then return the students that match that criteria. Constraint(s): The user may provide the wrong search criteria which could lead to them not getting the correct search result
Business rules applicable to this requirement	The student must have already created their account. The admin must be logged onto the system to be able to search.
Revision date and Revision number:	2021-04-06 Version 1.0
Criticality/Priority:	Low

Functional Requirement	Explanation
Requirement number:	1.22
Requirement name (use case name):	Maintain Student
Requirement short description:	The system must allow the admin to update or delete student's accounts and details.
Requirement detailed description and constraints:	The system must allow for the admin to access the student's account and delete the account or update the information that the account holds. If the admin decides to delete the account, that specific account will be removed from the database. If the user decides to



	update the account, the user should be able to change the information initially provided. The newly provided information will be checked to see if it is of the correct data type and then be saved to the database. • If the edit option is selected the all the details can be edited • If the delete option is selected the account and all its dependencies must be removed from the database.
Business rules applicable to this requirement	The student must have already created their account. The admin must be logged onto the system to be able to search.
Revision date and Revision number:	2021-04-06 Version 1.0
Criticality/Priority:	Medium

Functional Requirement	Explanation
Requirement number:	1.23
Requirement name (use case name):	Create Session Content Category
Requirement short description:	The system must allow the user to Create Session Content Categories for the sessions to be presented.
Requirement detailed description and constraints:	The system must allow for the admin Create Session Content Categories for the sessions to be presented. This used to categorise the live sessions presented under their specific categories.
Business rules applicable to this requirement	• The user must be an admin or tutor. • The user must be logged in to the system to be able to create.
Revision date and Revision number:	2021-04-06 Version 1.0
Criticality/Priority:	High

Functional Requirement	Explanation
------------------------	-------------



Requirement number:	1.24
Requirement name (use case name):	Search session content category
Requirement short description:	The system will allow the admin through its interface to search for all existing session content categories.
Requirement detailed description and constraints:	The system allows for session content categories created to be searched by the admin. The system will search for the session content categories by using specific search criteria. The system will then return the session content categories that match that criteria. Constraint(s): The user may provide the wrong search criteria which could lead to them not getting the correct search result
Business rules applicable to this requirement	The session content categories must have already been created. The admin must be logged onto the system to be able to search.
Revision date and Revision number:	2021-04-06 Version 1.0
Criticality/Priority:	Medium

Functional Requirement	Explanation
Requirement number:	1.25
Requirement name (use case name):	Maintain session content category
Requirement short description:	The system must allow the admin to update or delete session content categories
Requirement detailed description and constraints:	The system must allow for the admin to access the session content category and delete the session content category or update the information that the session content category holds. If the admin decides to delete the session content category, that specific category will be removed from the database. If the user decides to update the session content category, the user should be able to change the information initially provided. The newly provided information will



	be checked to see if it is of the correct data type and then be saved to the database. If the edit option is selected the all the details can be edited If the delete option is selected the session content category and all its dependencies must be removed from the database.
Business rules applicable to this requirement	The session content category must already be created. The admin must be logged onto the system to be able to search.
Revision date and Revision number:	2021-04-06 Version 1.0
Criticality/Priority:	Medium

Functional Requirement	Explanation
Requirement number:	1.26
Requirement name (use case name):	Create Course Content Category
Requirement short description:	The system must allow the user to Create Course Content Categories for the courses to be presented.
Requirement detailed description and constraints:	The system must allow for the admin Create Course Content Categories for the Courses to be presented. This used to categorise the course content under their specific categories.
Business rules applicable to this requirement	• The user must be an admin or tutor. • The user must be logged in to the system to be able to create.
Revision date and Revision number:	2021-04-06 Version 1.0
Criticality/Priority:	High

Functional Requirement	Explanation
Requirement number:	1.27
Requirement name (use case name):	Search course content category



Requirement short description:	The system will allow the admin through its interface to search for all existing course content categories.
Requirement detailed description and constraints:	The system allows for course content categories created to be searched by the admin. The system will search for the course content categories by using specific search criteria. The system will then return the course content categories that match that criteria. Constraint(s): The user may provide the wrong search criteria which could lead to them not getting the correct search result
Business rules applicable to this requirement	The course content categories must have already been created. The admin must be logged onto the system to be able to search.
Revision date and Revision number:	2021-04-06 Version 1.0
Criticality/Priority:	Medium

Functional Requirement	Explanation
Requirement number:	1.28
Requirement name (use case name):	Maintain course content category
Requirement short description:	The system must allow the admin to update or delete course content categories
Requirement detailed description and constraints:	The system must allow for the admin to access the course content category and delete the course content category or update the information that the course content category holds. If the admin decides to delete the course content category, that specific category will be removed from the database. If the user decides to update the course content category, the user should be able to change the information initially provided. The newly provided information will be checked to see if it is of the correct data type and then be saved to the database. If the edit option is selected the all the details can be edited



	If the delete option is selected the course content category and all its dependencies must be removed from the database.
Business rules applicable to this requirement	The course content category must already be created. The admin must be logged onto the system to be able to search.
Revision date and Revision number:	2021-04-06 Version 1.0
Criticality/Priority:	Medium

Functional Requirement	Explanation
Requirement number:	1.29
Requirement name (use case name):	Add course content
Requirement short description:	The system must allow the user upload all content that is related to the course to the system/database
Requirement detailed description and constraints:	• The system will allow the user to choose the upload option for a course in order to upload content. • Content that can be uploaded include: ○ course notes ○ Videos for that course
Business rules applicable to this requirement	• The user must be of type tutor or admin in order to add content to the course • The user must be logged in to the system to be able to add content. • The content must be in the correct format
Revision date and Revision number:	2021-04-06 Version 1.0
Criticality/Priority:	High

Functional Requirement	Explanation
------------------------	-------------



Requirement number:	1.30
Requirement name (use case name):	Search course content
Requirement short description:	The system must allow the user to search for uploaded course content that is linked to a specific course
Requirement detailed description and constraints:	• The system allows for the content uploaded to a specific course to be searched. • The system will search for the uploaded session content by using specific search criteria like date of upload and content type and title name. The system will then return the content that match that criteria. Constraint(s): • The user may provide the wrong search criteria which could lead to them not getting the correct search result
Business rules applicable to this requirement	• The user must be of type tutor or admin in order to search the content of the course • The user must be logged in to the system to be able to search for uploaded content
Revision date and Revision number:	2021-04-06 Version 1.0
Criticality/Priority:	Low

Functional Requirement	Explanation
Requirement number:	1.31
Requirement name (use case name):	Maintain course content
Requirement short description:	The system must allow the user to delete or update course content that is linked to a course.
Requirement detailed description and constraints:	The system must allow for the user to access the uploaded course content of a specific course to either delete the content or update the content of that course. If the user decides to delete the content, that specific content will be removed from the database. If the user decides to update the content, the user should be able to change the content and its description initially provided. The newly provided



	information will be checked to see if it is of the correct data type and then be saved to the database. • If the edit option is selected the following can be edited ○ Title ○ Video ○ notes If the delete option is selected the content and all its dependencies must be removed from the database.
Business rules applicable to this requirement	• The user must be of type tutor or admin in order to maintain content to the course • The user must be logged in to the system to be able to maintain content.
Revision date and Revision number:	2021-04-06 Version 1.0
Criticality/Priority:	Medium



6.2.2 Login Sub-system

Functional Requirement:	Explanation:
Requirement number:	2.1
Requirement name (use case name):	Login
Requirement short description:	The system will provide a user with an interface to login in to the system
Requirement detailed description and constraints:	The user will be required to enter their username and password to be able to log in to the system. The system will verify the entered information with information in the database about the specific user before granting the user access to the system. Constraint(s): None
Business rules applicable to this requirement	• The user must be registered on the system to be able to log in. • When a user is logged in all functionality assigned to their account must be available to them.
Revision date and Revision number:	2021-04-03 Version 1.0
Criticality/Priority:	High

Functional Requirement:	Explanation:
Requirement number:	2.2
Requirement name (use case name):	Logout
Requirement short description:	The system will provide a user with an interface to log out of the system
Requirement detailed description and constraints:	A system user needs to be able to navigate to the logout button option at any point in using the system. The system will prompt the user to confirm their decision to logout with a confirmation message. The user will be able to confirm and log out of the system.



	Constraint(s): None
Business rules applicable to this requirement	• The user must be logged in to the system to be able to log out. • Users should be logged out of the system automatically if they close the web browser.
Revision date and Revision number:	2021-04-03 Version 1.0
Criticality/Priority:	High

Functional Requirement:	Explanation:
Requirement number:	2.3
Requirement name (use case name):	Reset Password
Requirement short description:	The system will provide a user with an interface to reset their forgotten password currently used on the system
Requirement detailed description and constraints:	The user should be able to select the forgot/reset password button. The system should prompt the user to enter his/her username. The system will then send a reset password email to the email address linked with the username in the database. The user should be able to access the email and click the reset password link which will redirect them back to the system to enter and confirm a new password. Constraint(s): The user must have access to the email address linked in the user's profile for him/her to receive the email with the reset password link
Business rules applicable to this requirement	• The user should only be able to reset their own password. • The user must be registered on the system to be able to reset their password
Revision date and Revision number:	2021-04-03 Version 1.0
Criticality/Priority:	High

Functional Requirement:	Explanation:
-------------------------	--------------



Requirement number:	2.4
Requirement name (use case name):	Change password
Requirement short description:	The system will provide a user with an interface to change their current password used on the system.
Requirement detailed description and constraints:	The user should be able to select the change password button after logging in as a user. The system should prompt the user to enter his/her current password and new password. The system will verify the current password to the current users' information in the database. If the current password is correct and matches the current users' information in the database, the users' password will be updated/changed using the new password entered. Constraint(s): None
Business rules applicable to this requirement	• The user should only be able to change their own password. • The user must be registered on the system and also be logged in to be able to change their password
Revision date and Revision number:	2021-04-04 Version 1.0
Criticality/Priority:	High

Functional Requirement:	Explanation:
Requirement number:	2.5
Requirement name (use case name):	Add user role
Requirement short description:	The system will provide a user with an interface to add a new user role to the system
Requirement detailed description and constraints:	The system will allow for a new user role to be added by requiring a user role name and user role description. The system will validate that the user role given does not already exist on the system. After validating that the user role does not exist already, it is added to the database as a new user role. Constraint(s):



	None
Business rules applicable to this requirement	• The user must be registered on the system and logged in to the system to be able to add a new user role. • Only authorized users of the system will be able to add a new user role. • A user role cannot be added if it already exists on the system.
Revision date and Revision number:	2021-04-03 Version 1.0
Criticality/Priority:	High

Functional Requirement:	Explanation:
Requirement number:	2.6
Requirement name (use case name):	Search user role
Requirement short description:	The system will provide a user with an interface to search an existing user role on the system
Requirement detailed description and constraints:	The system will allow for a user role to be searched by prompting the user to enter the user role name. The system will return all user roles from the database that match the user role name criteria entered in the search. Constraint(s): • The user may provide the wrong search criteria which could lead to them not getting the correct search result
Business rules applicable to this requirement	• The user must be registered on the system and logged in to the system to be able to search a user role. • A user role cannot be searched if it does not exist on the system. • Only authorized users can search for a user role.
Revision date and Revision number:	2021-04-04 Version 1.0
Criticality/Priority:	Low

Functional Requirement:	Explanation:
Requirement number:	2.7



Requirement name (use case name):	Maintain user role
Requirement short description:	The system will provide a user with an interface to update or delete an existing user role on the system
Requirement detailed description and constraints:	The system will allow for a user role to be updated or deleted after the user role was searched for (invoke 2.6 Search user role). The system will give the option to update or delete the specific searched user role. If a user selects the delete option the user role will be removed from the database after confirmation. If a user selects the update option the user should be able to change the user role name or user role description and the new information should be saved to the database after confirmation. Constraint(s): None
Business rules applicable to this requirement	• New /updated details entered should correspond to the required format. • The user role must already exist on the system to be deleted/updated. • Only authorized users can delete/ update a user role. • The user must be registered on the system and logged in to the system to be able to update/delete a user role.
Revision date and Revision number:	2021-04-04 Version 1.0
Criticality/Priority:	Medium



6.2.3 Student Sub-system

Functional Requirement	Explanation
Requirement number:	3.1
Requirement name (use case name):	Register student
Requirement short description:	The system must allow a user to provide their personal details and thus create them an account so they can use the system.
Requirement detailed description and constraints:	• The details provided by the student should be entered on the webpage and then saved under their new account. • The user should provide the following Details: ○ first name ○ surname ○ email address, ○ phone number ○ university/institution they attend ○ subject module ○ username ○ password • The account will be assigned a student role. • All data entered by the student should be validated to ensure the correct data was provided. • Personal details provided such as phone number and email address should be unique to the student and in turn the newly created account will be saved in the database.
Business rules applicable to this requirement	Only students who do not already exist on the system can register for an account.
Revision date and Revision number:	2021-04-04 Version 1.0
Criticality/Priority:	High



Functional Requirement	Explanation
Requirement number:	3.2
Requirement name (use case name):	Maintain Student
Requirement short description:	The system must allow a registered student to update or delete their account.
Requirement detailed description and constraints:	The system must allow for a registered student to access their account and either delete their account or update the information that their account holds. If the student decides to delete the account the specific account will be removed from the database. If the student decides to update the account, the student should be able to change their personal details and thus the new information provided must first be validated to check that it is of the correct data type and in turn it should be saved to the database. • If the edit option is selected the following can be edited ○ First name ○ Surname ○ Email address ○ Phone number ○ Institution ○ Module ○ Username ○ Password • If the delete option is selected the account and all its dependencies must be removed from the database.
Business rules applicable to this requirement	• Only students that are registered on the system can maintain their account • The edited details must be in the correct format before they are saved.
Revision date and Revision number:	2021-04-04 Version 1.0
Criticality/Priority:	Medium



Functional Requirement	Explanation
Requirement number:	3.3
Requirement name (use case name):	Vote for group session
Requirement short description:	The system must allow a registered student to view the possible session times and then vote on which time they can attend
Requirement detailed description and constraints:	The system allows the student to view the proposed session/lesson timeslots and vote on which one they can attend. The students are presented with the option to click on the vote button which then record their details such as their name and email address under that specific proposed timeslot. • The poll that the students vote on has the following information: ○ Lesson Name ○ Tutor Name ○ Session content ○ Date and time. The students have a limit on how many proposed timeslots they can vote for, thus they can only vote to attend two different timeslots and in turn their ability to vote again becomes disabled.
Business rules applicable to this requirement	• The student must be logged in on the system. • They must have purchased the Noob/standard tutoring package in order to vote. • They can only vote twice and no more.
Revision date and Revision number:	2021-04-04 Version 1.0
Criticality/Priority:	High



Functional Requirement	Explanation
Requirement number:	3.4
Requirement name (use case name):	Register for group session
Requirement short description:	After the voting has conclude, the system must allow a student to register for the highest voted/winning session in order to show that they will attend.
Requirement detailed description and constraints:	- Once Voting for the session has been concluded, the session with the highest vote will be scheduled and go ahead. Thus, students must register for the finalized timeslot/session to show the tutor that they are able to attend. This is done by clicking on the register option and the student details such as name and email will be saved under that registration. - Only students that have registered for the session will be given the link for the session on zoom via email and thus students who do not have the correct details will not have access.
Business rules applicable to this requirement	- The student must be logged in on the system. - They must have purchased the Noob/standard tutoring package in order to vote.
Revision date and Revision number:	2021-04-04 Version 1.0
Criticality/Priority:	High

Functional Requirement	Explanation
Requirement number:	3.5
Requirement name (use case name):	Make individual booking
Requirement short description:	The system must allow a student who has purchased the Pro tutoring package to book an individual session with a tutor once a week.
Requirement detailed description and constraints:	The student navigates to that calendar page in order to view what timeslots are available to be booked.



	- The student chooses a time slot that fits their schedule and proceeds to write a description of what they would like to cover in the session and then proceeds to choose the book option. - The time slot will then be booked and no longer available and in turn both the student and tutor will receive an email confirming the session has been booked
Business rules applicable to this requirement	- The student must be logged in on the system. - They must have purchased the Pro tutoring package to book a private session - The students cannot book more than one timeslot a week
Revision date and Revision number:	2021-04-04 Version 1.0
Criticality/Priority:	High

Functional Requirement	Explanation
Requirement number:	3.6
Requirement name (use case name):	Search individual booking
Requirement short description:	The system will allow the student through its interface to search for all existing individual sessions that the he/she has booked.
Requirement detailed description and constraints:	- The system allows for the individual sessions booked by the student to be searched. - The system will search for the individual session by using specific search criteria like date and time and the tutor's name. The system will then return the booked sessions that match that criteria. Constraint(s): - The student may provide the wrong search criteria which could lead to them not getting the correct search result
Business rules applicable to this requirement	The student must have purchased the Pro package in order to view bookings .



	The student must be logged onto the system to be able to search.
Revision date and Revision number:	2021-04-04 Version 1.0
Criticality/Priority:	Low

Functional Requirement	Explanation
Requirement number:	3.7
Requirement name (use case name):	Maintain individual booking
Requirement short description:	The system must allow the student who has individual bookings to update or delete their individual bookings.
Requirement detailed description and constraints:	The system must allow for the student to access their current bookings and either delete the booking or update the information that the booking holds. If the student decides to delete the booking, that specific booking will be removed from the database. If the student decides to update the booking, the student should be able to change the information initially provided. The newly provided information will be checked to see if it is of the correct data type and then be saved to the database. - If the edit option is selected the following can be edited - Date and time - Content to be discussed in the session - If the delete option is selected the booking and all its dependencies must be removed from the database.
Business rules applicable to this requirement	- The student must have purchased the Pro package in order to view and maintain bookings. - The student must be logged onto the system to be able to search. - The student must have existing bookings. - Timeslots cannot be updated/changed or deleted 24hours before the booked session.
Revision date and Revision number:	2021-04-04 Version 1.0



Criticality/Priority:	Medium
-----------------------	--------

Functional Requirement	Explanation
Requirement number:	3.8
Requirement name (use case name):	Create Message
Requirement short description:	The system must allow the student to compose a message and send it to a tutor.
Requirement detailed description and constraints:	The system must allow for the student to type out a message and send it to a specific tutor. This is achieved by selecting which tutor the student wishes to message and the typing out the content of the message which in turn is validated.
Business rules applicable to this requirement	• The student must have purchased the Pro or Noob package in order send a message • The student must be logged in to the system to be able to create a message.
Revision date and Revision number:	2021-04-04 Version 1.0
Criticality/Priority:	High

Functional Requirement	Explanation
Requirement number:	3.9
Requirement name (use case name):	Search Message
Requirement short description:	The system must allow the student to search for a message that he/she sent or received.
Requirement detailed description and constraints:	• The system allows for the messages sent and received by the student to be searched. • The system will search for the messages by using specific search criteria like date and time that the messages were sent and the recipient or sender's name. The system will then return the messages that match that criteria.



	Constraint(s): • The student may provide the wrong search criteria which could lead to them not getting the correct search result
Business rules applicable to this requirement	• The student must have purchased the Pro or Noob package in order to view messages • The student must be logged in to the system to be able to search.
Revision date and Revision number:	2021-04-04 Version 1.0
Criticality/Priority:	Low

Functional Requirement	Explanation
Requirement number:	3.10
Requirement name (use case name):	Maintain Message
Requirement short description:	The system must allow the student to delete messages that he/she sent or received
Requirement detailed description and constraints:	• The system allows for the messages sent and received by the student to be deleted once the student is finished reading them or they wish to no longer have them in their records. • The system will provide a list of the messages sent or received and then the student can delete the specific message by choosing the delete option. If the message was searched for (invoke 3.9 search message), the system will give the option to delete the specifically searched message. • The message will then be deleted from the database.
Business rules applicable to this requirement	• The student must have purchased the Pro or Noob package in order to view messages • The student must be logged in to the system to be able to search.
Revision date and Revision number:	2021-04-04 Version 1.0



Criticality/Priority:	Medium
-----------------------	--------

Functional Requirement	Explanation
Requirement number:	3.11
Requirement name (use case name):	Add to basket
Requirement short description:	The system must allow the student to add optional courses to a basket in order to purchase them
Requirement detailed description and constraints:	• The system allows for students to purchase optional courses by adding them to their basket when choosing the buy option. • The student will go to the courses page and scroll through a variety of courses that are on the system and select which one they want to add to their basket. • The courses have a description of the content taught and a price. When the student wants to purchase the course, they choose the buy option which then adds that specific course to their basket.
Business rules applicable to this requirement	• The student must be logged onto the system to be able to add a course to their basket.
Revision date and Revision number:	2021-04-04 Version 1.0
Criticality/Priority:	High

Functional Requirement	Explanation
Requirement number:	3.12
Requirement name (use case name):	Remove item/(s) from basket
Requirement short description:	The system must allow the student to remove courses from their basket that they no longer want
Requirement detailed description and constraints:	• The system will allow the student to view all the courses he or she chose the buy option for in their basket and in turn this will allow the student to choose the remove option for



	a specific course that exists inside their basket. The system will then show all remaining courses in the basket.
Business rules applicable to this requirement	• There must be items existing in the students basket. • The student must be logged in to the system to be able to remove items from their basket
Revision date and Revision number:	2021-04-04 Version 1.0
Criticality/Priority:	High

Functional Requirement	Explanation
Requirement number:	3.13
Requirement name (use case name):	View basket
Requirement short description:	The system must allow the student to view their basket and all items inside the basket.
Requirement detailed description and constraints:	• The system will allow the student to view his/her basket. • When the student views the basket, they are presented with either an empty basket if they have not added any courses to the basket or a list of all the courses they chose to add. • Inside the basket the following details will be provided: ○ Course name ○ Price of course ○ Subtotal of all courses in the basket • The basket will also have a checkout and empty all option.
Business rules applicable to this requirement	• The student must be logged onto the system to be able view their basket
Revision date and Revision number:	2021-04-04 Version 1.0
Criticality/Priority:	High



Functional Requirement	Explanation
Requirement number:	3.14
Requirement name (use case name):	Checkout
Requirement short description:	The system must allow the student to checkout their basket.
Requirement detailed description and constraints:	The system will allow the student to view his/her basket and when the student decides to purchase the items, they choose the checkout option. This then clears the basket and redirects the student to the Paystack payment gate in order to make payment. On completion all items will be saved under the student's account.
Business rules applicable to this requirement	- There must be items existing in the students basket. - The student must be logged in to the system to be able to checkout.
Revision date and Revision number:	2021-04-04 Version 1.0
Criticality/Priority:	High

Functional Requirement	Explanation
Requirement number:	3.15
Requirement name (use case name):	Create Feedback
Requirement short description:	The system must allow the student to provide feedback on a specific tutor and their overall experience that they have had with that tutor and the system.
Requirement detailed description and constraints:	The system must allow for the student to click on the feedback option. The feedback option will use a star rating criterion where five stars represents a standard of excellence while one star represents a poor standard, the system also will allow the student to provide a description of their opinions and possible solutions to their experiences. The purpose of the feedback form is to rate a specific tutor and for the student to indicate if they have experienced any issues with that specific tutor.



	• The form will allow for the following to be rated in terms of a tutor: ○ The friendliness of the tutor ○ The teaching ability of the tutor ○ The punctuality/timeliness of the tutor • The feedback form will also allow the student to rate the system as a whole. • The form will also allow the student to write a brief description which will be based on general feedback they have for that specific tutor. • All inputs will be validated to see if they are of the correct data type and not null and in turn the feedback form will be saved to the database.
Business rules applicable to this requirement	• The student must have purchased the Pro or Noob package in order to provide • The student must be logged in to the system to be able to create a feedback form.
Revision date and Revision number:	2021-04-04 Version 1.0
Criticality/Priority:	High

Functional Requirement	Explanation
Requirement number:	3.16
Requirement name (use case name):	Search Feedback
Requirement short description:	The system must allow the student to search for Feedback forms that he/she sent has created.
Requirement detailed description and constraints:	• The system allows for the forms that the student has created to be searched. • The system will search for the feedback forms by using specific search criteria most recent/least recent or based on a Tutors name. The system will then return the messages that match that criteria.



	Constraint(s): • The student may provide the wrong search criteria which could lead to them not getting the correct search result
Business rules applicable to this requirement	• The student must have purchased the Pro or Noob package in order to search for their feedback forms. • The student must be logged in to the system to be able to search.
Revision date and Revision number:	2021-04-04 Version 1.0
Criticality/Priority:	Low

Functional Requirement	Explanation
Requirement number:	3.17
Requirement name (use case name):	Maintain Feedback
Requirement short description:	The system must allow the student to update or delete feedback forms that he/she sent has created
Requirement detailed description and constraints:	• The system allows for the student to edit/update or delete their existing feedback forms. • The system will provide a list of all the feedback forms created by the student and will allow them to either choose the update or delete option. If the delete option is selected the system will ask the student to confirm if they want to delete the form if so the feedback form is deleted from the database. If the student selects the update option the system provides the form and the student can edit the existing data. • The existing data includes: ○ The friendliness of the tutor ○ The teaching ability of the tutor ○ The punctuality/timeliness of the tutor ○ The overall experience they have received



	○ Description • . If the Feedback form was searched for (invoke 3.16 search feedback), the system will give the option to delete the specifically searched feedback form.
Business rules applicable to this requirement	• The student must have purchased the Pro or Noob package in order to maintain the feedback form. • The student must be logged in to the system to be able to maintain the feedback form.
Revision date and Revision number:	2021-04-04 Version 1.0
Criticality/Priority:	Medium



6.2.4 Tutor Sub-system

Functional Requirement:	Explanation:
Requirement number:	4.1
Requirement name (use case name):	Apply to become a tutor
Requirement short description:	The system must allow a user to apply to become a tutor without being a registered or logged in user on the system. The login interface should have 'Apply to become a tutor' as an option
Requirement detailed description and constraints:	The system will allow a user to select the 'Apply to become a tutor' option on the login interface, thereafter the system will redirect the user to a page where they can submit their application to become a tutor. Constraint(s): • Once an application is submitted it cannot be cancelled by the user.
Business rules applicable to this requirement	• To submit an application a user does not have to be registered or logged in on the system
Revision date and Revision number:	2021-04-04 Version 1.0
Criticality/Priority:	High

Functional Requirement:	Explanation:
Requirement number:	4.2
Requirement name (use case name):	Create group session vote
Requirement short description:	The system must allow a tutor user role to create a voting poll for students to vote for three different session timeslots for group classes
Requirement detailed description and constraints:	The system will allow a tutor with the option on the home interface to set up voting polls for different group session options. The voting polls will be made available for students to vote for before a decision is made on which timeslot will be used for a group session. Once the voting polls are set up, an automatic email is



	sent out to all registered students of that specific module to notify them to go and vote. Constraint(s): • A tutor can only set up voting polls for three possible sessions
Business rules applicable to this requirement	• Only registered students can vote by use of this pole • Only tutors can create the voting poles for the three potential sessions based on their schedules
Revision date and Revision number:	2021-04-04 Version 1.0
Criticality/Priority:	High

Functional Requirement:	Explanation:
Requirement number:	4.3
Requirement name (use case name):	Search group session vote
Requirement short description:	The system must allow a tutor/admin user role to search a voting poll that was created for different session timeslots for group classes
Requirement detailed description and constraints:	The system will allow the created voting polls to be searched for by prompting the tutor/admin user to enter the poll time and date. The system will return all polls from the database that match the poll time and date criteria entered in the search. Constraint(s): • The tutor/admin user may provide the wrong search criteria which could lead to them not getting the correct search result
Business rules applicable to this requirement	• Only tutor and admin user roles can search voting polls
Revision date and Revision number:	2021-04-04 Version 1.0
Criticality/Priority:	Low

Functional Requirement:	Explanation:
-------------------------	--------------



Requirement number:	4.4
Requirement name (use case name):	Maintain group session vote
Requirement short description:	The system must allow a tutor/admin user role to update/delete a voting poll that was created for different session timeslots for group classes
Requirement detailed description and constraints:	The system will allow a created voting poll to be updated or deleted after it was searched for (invoke 4.3 Search group session vote). The system will give the option to update or delete the specifically searched for voting poll. If a admin/tutor user selects the delete option the specific voting poll will be removed from the database after confirmation. If a admin/tutor user selects the update option the user should be able to change the specific voting poll's time or date and the new information should be saved to the database after confirmation. Constraint(s): None
Business rules applicable to this requirement	• Only tutor and admin user rolls can update/delete a voting poll
Revision date and Revision number:	2021-04-04 Version 1.0
Criticality/Priority:	Medium

Functional Requirement:	Explanation:
Requirement number:	4.5
Requirement name (use case name):	Tutor confirms group voted session
Requirement short description:	The system should allow the tutor to confirm the session that received the most votes by students
Requirement detailed description and constraints:	The system will allow a tutor to confirm which session out of the three options created received the most votes. After confirming the winning session automatic emails are sent out to all students registered on the system for that module informing them about the set date and time for the session. The email will also contain a link for the students to follow to go and register for the set session.



	Constraint(s): None
Business rules applicable to this requirement	• Only the tutor who created the voting polls will be able to confirm the winning one • All students registered for the module, even if they did not vote for the winning timeslot, receives the email confirming the winning session date and time.
Revision date and Revision number:	2021-04-04 Version 1.0
Criticality/Priority:	High

Functional Requirement:	Explanation:
Requirement number:	4.6
Requirement name (use case name):	Create group session registration
Requirement short description:	The system must allow a tutor user role to create registration for the winning group session, where all students can go and register for the session
Requirement detailed description and constraints:	The system will allow a tutor with the option on the home interface to set up a registration area for a specific session for students to go and register for. After following the link in the automatic emails sent to all registered students, students will be redirected to this registration area where they can register for the set session. Constraint(s): None
Business rules applicable to this requirement	• A tutor must be logged on to the system to set up a registration area • The registration period will have a cut off date/time
Revision date and Revision number:	2021-04-04 Version 1.0
Criticality/Priority:	High



Functional Requirement:	Explanation:
Requirement number:	4.7
Requirement name (use case name):	Search group session registration
Requirement short description:	The system must allow a tutor/admin user role to search registration where students register for the winning group session.
Requirement detailed description and constraints:	The system will allow the registration area to be searched for by prompting the tutor/admin user to enter in the winning session's time and date that is being registered for. The system will return all registration areas from the database that match the entered session time and date criteria in the search. Constraint(s): • The tutor/admin user may provide the wrong search criteria which could lead to them not getting the correct search result
Business rules applicable to this requirement	• A tutor/admin user must be logged on to the system to search the registration area
Revision date and Revision number:	2021-04-04 Version 1.0
Criticality/Priority:	Low

Functional Requirement:	Explanation:
Requirement number:	4.8
Requirement name (use case name):	Maintain group session registration
Requirement short description:	The system must allow a tutor/admin user role to update/delete registration where students register for the winning group session
Requirement detailed description and constraints:	The system will allow the registration area to be updated or deleted after it was searched for (invoke 4.7 Search group session registration). The system will give the option to update or delete the specifically searched for registration area. If a admin/tutor user selects the delete option the specific registration area will be removed from the database after confirmation. If a admin/tutor user selects the update option the user



	should be able to change the specific registration area's information by removing/adding student participants and the new information should be saved to the database after confirmation. Constraint(s): None
Business rules applicable to this requirement	• A tutor/admin user must be logged on to the system to be able to maintain a registration area
Revision date and Revision number:	2021-04-04 Version 1.0
Criticality/Priority:	Medium

Functional Requirement:	Explanation:
Requirement number:	4.9
Requirement name (use case name):	Create session
Requirement short description:	The system will provide the tutor with an interface option to create a session that won the most votes and that students registered for
Requirement detailed description and constraints:	The system will allow a tutor with the interface option to create a session that won the most votes and that students registered for. An automatic email is sent out to all students that registered for the session that contains a link that will redirect the students to the video chat session. Constraint(s): None
Business rules applicable to this requirement	• A tutor must be logged on to the system in order to create a session
Revision date and Revision number:	2021-04-04 Version 1.0
Criticality/Priority:	High



Functional Requirement:	Explanation:
Requirement number:	4.10
Requirement name (use case name):	Search session
Requirement short description:	The system must allow a tutor/admin user role to search a created session.
Requirement detailed description and constraints:	The system will allow the created session to be searched for by prompting the tutor/admin user to enter in the session time and date. The system will return all created sessions from the database that match the entered session time and date criteria in the search. Constraint(s): • The tutor/admin user may provide the wrong search criteria which could lead to them not getting the correct search result
Business rules applicable to this requirement	• A tutor/admin user must be logged on to the system to search the created session
Revision date and Revision number:	2021-04-04 Version 1.0
Criticality/Priority:	Low

Functional Requirement:	Explanation:
Requirement number:	4.11
Requirement name (use case name):	Maintain session
Requirement short description:	The system must allow a tutor/admin user role to update/delete a session
Requirement detailed description and constraints:	The system will allow the session to be updated or deleted after it was searched for (invoke 4.10 Search session). The system will give the option to update or delete the specifically searched for session. If a admin/tutor user selects the delete option the specific session will be removed from the database after confirmation. If a admin/tutor user selects the update option the user should be able to change the specific session's



	information by changing the time or date and the new information should be saved to the database after confirmation. Constraint(s): None
Business rules applicable to this requirement	• A tutor/admin user must be logged on to the system to be able to maintain a session
Revision date and Revision number:	2021-04-04 Version 1.0
Criticality/Priority:	Medium

Functional Requirement	Explanation
Requirement number:	4.12
Requirement name (use case name):	Take attendance
Requirement short description:	The system must allow the Tutor to take attendance of all the students present in the session.
Requirement detailed description and constraints:	• The system will allow the tutor to access a list of registered students for that session. Tutors will then let the system know who is present through the form of a list. • An attendance list will be needed for each session.
Business rules applicable to this requirement	• Every student present must be registered. • Only registered students can join a session. • A tutor needs to fill out an attendance form before every session ends
Revision date and Revision number:	2021-04-04 Version 1.0
Criticality/Priority:	High



Functional Requirement	Explanation
Requirement number:	4.13
Requirement name (use case name):	Search attendance
Requirement short description:	The system must allow the tutor to search for previous attendance lists.
Requirement detailed description and constraints:	• The system allows for the attendance lists from previous sessions to be searched for. • The system will search for the previous attendance lists by using specific search criteria like date of upload and tutor responsible and title name. The system will then return the content that match that criteria. Constraint(s): • The tutor may provide the wrong search criteria which could lead to them not getting the correct search result
Business rules applicable to this requirement	• The user must be of type tutor in order to search the content of the past session • The tutor must be logged in to the system to be able to search for uploaded content.
Revision date and Revision number:	2021-04-04 Version 1.0
Criticality/Priority:	Low

Functional Requirement	Explanation
Requirement number:	4.14
Requirement name (use case name):	Maintain attendance list
Requirement short description:	The system must allow the tutor to delete or update session attendance lists that is linked to a specific past session.



Requirement detailed description and constraints:	The system must allow for the tutor to access the attendance lists of a specific past session in order to either delete the attendance lists or update the attendance lists of that past session. If the tutor decides to update the attendance lists, the tutor should be able to change the attendance lists. The newly provided information will be checked to see if it is of the correct data type and then be saved to the database. • If the edit option is selected the following can be edited ○ Student present or not Boolean • If the delete option is selected the content and all its dependencies must be removed from the database.
Business rules applicable to this requirement	• The user must be of type tutor to maintain content to the past session. • The Tutor must be logged in to the system to be able to maintain content.
Revision date and Revision number:	2021-04-04 Version 1.0
Criticality/Priority:	Medium

Functional Requirement	Explanation
Requirement number:	4.15
Requirement name (use case name):	Add group session content
Requirement short description:	The system must allow the Tutor to upload all content that is related to the past session to the system/database.
Requirement detailed description and constraints:	• The system will allow the tutor to choose the upload option for a past session in order to upload content that was discussed in the session to that specific session. • Content that can be uploaded to the past session include: ○ Session notes ○ Recordings of the session



	All content uploaded will have a title.
Business rules applicable to this requirement	- The user must be of type tutor in order to add content to the past session. - The tutor must be logged in to the system to be able to add content. - The content must be in the correct format.
Revision date and Revision number:	2021-04-04 Version 1.0
Criticality/Priority:	High

Functional Requirement	Explanation
Requirement number:	4.16
Requirement name (use case name):	Search group session content
Requirement short description:	The system must allow the tutor to search for uploaded session content that is linked to a specific past session.
Requirement detailed description and constraints:	- The system allows for the content uploaded to a specific past session to be searched. - The system will search for the uploaded session content by using specific search criteria like date of upload and content type and title name. The system will then return the content that match that criteria. Constraint(s): - The tutor may provide the wrong search criteria which could lead to them not getting the correct search result
Business rules applicable to this requirement	- The user must be of type tutor to search the content of the past session - The tutor must be logged in to the system to be able to search for uploaded content.
Revision date and Revision number:	2021-04-04 Version 1.0
Criticality/Priority:	Low



Functional Requirement	Explanation
Requirement number:	4.17
Requirement name (use case name):	Maintain group session content
Requirement short description:	The system must allow the tutor to delete or update session content that is linked to a specific past session.
Requirement detailed description and constraints:	The system must allow for the tutor to access the uploaded session content of a specific past session in order to either delete the content or update the content of that past session. If the tutor decides to delete the content, that specific content will be removed from the database. If the tutor decides to update the content, the tutor should be able to change the content and its description initially provided. The newly provided information will be checked to see if it is of the correct data type and then be saved to the database. • If the edit option is selected the following can be edited ○ Title ○ Video recording ○ notes • If the delete option is selected the content and all its dependencies must be removed from the database.
Business rules applicable to this requirement	• The user must be of type tutor in order to maintain content to the past session. • The Tutor must be logged in to the system to be able to maintain content.
Revision date and Revision number:	2021-04-04 Version 1.0
Criticality/Priority:	Medium

Functional Requirement	Explanation
Requirement number:	4.18



Requirement name (use case name):	Search Student
Requirement short description:	The system must allow the Tutor to search for a registered student on the system.
Requirement detailed description and constraints:	- The system allows for students registered on the system to be searched and viewed by the tutors. - The system will search for the student by using specific search criteria such as the student first name, email address and module. The system will then return the content that match that criteria. Constraint(s): - The tutor may provide the wrong search criteria which could lead to them not getting the correct search result
Business rules applicable to this requirement	- The user must be of type tutor in order to search for a registered student - The tutor must be logged in to the system to be able to search for a student.
Revision date and Revision number:	2021-04-04 Version 1.0
Criticality/Priority:	Low

Functional Requirement	Explanation
Requirement number:	4.19
Requirement name (use case name):	Create individual timeslots
Requirement short description:	The system must allow the tutor create timeslots for the week when he or she is able to have private sessions.
Requirement detailed description and constraints:	- The system will allow the tutor to select a specific day and choose a time in that day when he/she is able to host an individual session. - When creating a timeslot, the tutor chooses a period that is no longer than an hour between 7:00 am and 20:00 pm on a specific day which is then saved to his or her calendar.



Business rules applicable to this requirement	• The user must be of type tutor in order to create a timeslot • The tutor must be logged in to the system to be able to create a timeslot. • Timeslots that are the same cannot be scheduled more than once on the same day.
Revision date and Revision number:	2021-04-04 Version 1.0
Criticality/Priority:	High

Functional Requirement	Explanation
Requirement number:	4.20
Requirement name (use case name):	Search Timeslot
Requirement short description:	The system must enable the user to be able to search for an already created timeslot for the week when the user can have private sessions
Requirement detailed description and constraints:	• The system will allow the user to search for a specific day that they have created a timeslot or multiple possible timeslots when they can host a private session for that particular day. • The system filters through the created sessions for that particular user (tutor) on their personal calendar. • When timeslots have been found for a particular day, the system will display the relevant timeslots based on input search criteria.
Business rules applicable to this requirement	• Only a tutor has access to the search timeslot functionality. • A user cannot search for a timeslot if the user is not logged into the system. • A timeslot cannot be displayed based on a specific search criterion if the timeslot did not already exist before the search was performed.
Revision date and Revision number:	2021-04-05 Version 1.0
Criticality/Priority:	Medium



Functional Requirement	Explanation
Requirement number:	4.21
Requirement name (use case name):	Maintain Timeslot
Requirement short description:	The system must allow the user to maintain (update or delete) any timeslots that they made available to host an individual session.
Requirement detailed description and constraints:	• The system first calls the “search timeslot” functionality to allow a tutor to search for a timeslot, that they have already created, to be able to alter or delete the timeslot • After the user searched for a timeslot and a timeslot was found, the system must allow the tutor to change the timing of the session (e.g. from 8-9 am to 10-11am) or delete the session completely.
Business rules applicable to this requirement	• A timeslot must first be searched for and a relevant match must be provided before a timeslot can be altered or removed • Only a tutor can maintain (update or delete) a specific timeslot • A tutor can only maintain a timeslot that they have created (no other tutors) • A tutor must be logged into the system to have access to this functionality.
Revision date and Revision number:	2021-04-05 Version 1.0
Criticality/Priority:	Low

Functional Requirement	Explanation
Requirement number:	4.22
Requirement name (use case name):	Maintain Tutor
Requirement short description:	The system must allow a user (tutor) to be able to update their personal profile on the system or allow them to delete their profile.



Requirement detailed description and constraints:	• The system must allow the user to navigate to a button that allows them to change the details on their personal tutor profile. • The system must provide for fields such as: Email, modules (that a tutor will tutor for), name, surname, banking details and cell phone number. • When the user has made their changes and confirms them, the system must first check if the entered/alterd/deleted data is in the correct format and if it is correct, the system must save these changes to the database
Business rules applicable to this requirement	• A tutor must be logged into the system to have access to this functionality. • A tutor must exist within the system database before their personal details can be altered or changed. • A tutor can only alter or delete details on their own profile, not any other tutor or any other student profiles. • Only administrators can delete a whole tutor's profile from the system (if the tutor no longer tutors at the business for example)
Revision date and Revision number:	2021-05-04 Version 1.0
Criticality/Priority:	Medium

Functional Requirement	Explanation
Requirement number:	4.23
Requirement name (use case name):	Create Resource Categories
Requirement short description:	The system must allow the user to create resource categories where under content will be added that is relevant to that specific category
Requirement detailed description and constraints:	• The system must allow the user to click on a button to add a new resource category ("Recordings" for explanation purposes) • The system will first perform a validation check to ensure that a category is not created that already exists within the database.



	• If the validation test was successful, the system will then capture the newly created resource category and add it to the database. • Content such as “Recording week 1” will be added under the “Recordings” resource category for this example.
Business rules applicable to this requirement	• Only a tutor must be allowed to create a resource category. • A tutor must be logged into the system to perform this action. • A category cannot be added if it already exists within the system. • A tutor can only add a resource category to the module that he tutors, no other modules (A tutor for INF 272 cannot add a resource category for INF 354)
Revision date and Revision number:	2021-05-04 Version 1.0
Criticality/Priority:	High

Functional Requirement	Explanation
Requirement number:	4.24
Requirement name (use case name):	Search Resource Categories
Requirement short description:	The system must allow the user to search for an already existing resource category based on a certain criterion
Requirement detailed description and constraints:	• The system must allow the user to click on a button to list the current resource categories found within the database with an input field that allows the user to sort through the categories to find the resource category he/she wishes to make changes to. • After the search input has been validated by a set of input rules and restrictions, the system will retrieve and display the relevant categories based on the search requirements of the tutor.



	A list of resource categories will be displayed with the name of the category and an option to delete or edit (update) the category.
Business rules applicable to this requirement	- The tutor must be logged in to the system. - The search input field must be validated before a search can be performed. - A tutor can only search for resource categories relevant to the module that they tutor for.
Revision date and Revision number:	2021-05-04 Version 1.0
Criticality/Priority:	Low

Functional Requirement	Explanation
Requirement number:	4.25
Requirement name (use case name):	Maintain Resource Categories
Requirement short description:	The system must allow a user (tutor) to alter the data of a resource category or have the option to delete the category completely from their list of resource categories for their specific module
Requirement detailed description and constraints:	- After the user searched for a category and the system retrieves and displays the results (call requirement number 4.26) , the user can then either click on the “delete” icon or the “edit” icon to change or delete details of a category. - After necessary changes are made to the chosen category, the user confirms his/her actions, the system validates changes and then updates the database accordingly.
Business rules applicable to this requirement	- A tutor must be logged into the system. - Only a tutor can change or delete resource categories that relates to the module that he tutors for. - A resource category can only be updated or deleted after it is : Either searched for by the tutor or it is displayed in a list format to the tutor.
Revision date and Revision number:	2021-05-04 Version 1.0
Criticality/Priority:	Medium



Functional Requirement	Explanation
Requirement number:	4.26
Requirement name (use case name):	Create Resource
Requirement short description:	The system must allow the user to add a new resource underneath a specific resource category
Requirement detailed description and constraints:	• The system must allow a tutor to make use of a button to add new resources (content) to a resource category it relates to. • Resources can be notes, assessments, tutorials, external links, articles, homework, recordings etc. • The system must then make sure that the resources that the user wants to create or add to the system, is in a format readable and executable by the system before it creates the resource and add it to the system. • When the tutor then confirms their newly created resource, the system will then add the resource to the database.
Business rules applicable to this requirement	• A tutor must be logged into the system. • A tutor can only add resources to the module they are specifically tutoring for. • A resource must be assigned to a resource category (it cannot not have a category associated to it) • Resources added must be in a format executable and manageable by the system before it can be accepted and added to the database
Revision date and Revision number:	2021-05-04 Version 1.0
Criticality/Priority:	High

Functional Requirement	Explanation
Requirement number:	4.27



Requirement name (use case name):	Search Resource
Requirement short description:	The system must allow the tutor to search for a resource that they have previously added to the database. (for removal or update purposes)
Requirement detailed description and constraints:	• The system must allow the tutor to search for a specific resource or group of resources either based on their resource category, resource name, date uploaded, file format or file size. • After the user enters their search input or specifications, the system will validate the search input and display the relevant resources based upon search criteria.
Business rules applicable to this requirement	• A tutor must be logged into the system. • Resources must be displayed in the fashion as stipulated by the user's search input requirements. • Only a tutor can search for a resource for the purpose of deleting or updating it.
Revision date and Revision number:	2021-05-04 Version 1.0
Criticality/Priority:	Medium

Functional Requirement	Explanation
Requirement number:	4.28
Requirement name (use case name):	Maintain Resource
Requirement short description:	The system must allow the user to update already existing resources or delete them completely from the database.
Requirement detailed description and constraints:	• After a list of resources is displayed by the system based on user search input, the system must allow the user to either delete a resource or update the contents thereof. • When the user wants to update the resource, the system must prompt the user with the appropriate input fields to perform a successful update. • After the input fields are validated, the system then captures the updates to the database



Business rules applicable to this requirement	• Tutor must be logged into the system. • A list of resources must first be displayed before a resource can be deleted or updated. • A tutor can only delete or update resources relevant to the specific module that they tutor for. • Only tutors can make use of this functionality.
Revision date and Revision number:	2021-05-04 Version 1.0
Criticality/Priority:	Medium

Functional Requirement	Explanation
Requirement number:	4.29
Requirement name (use case name):	Create message
Requirement short description:	The system must allow a tutor to compose a message and send it to a student/group of students
Requirement detailed description and constraints:	• The system must allow a tutor to search for a specific student or target group of students and allow the tutor to type out a message, which the content of the message is in correlation to the module or administrative matters, send and confirm the message to a specific student or group of students.
Business rules applicable to this requirement	• The user must be a tutor in order for them to send a message to a student. • A message must be addressed to a specific student or group. • The user must be logged into the system.
Revision date and Revision number:	2021-05-04 Version 1.0
Criticality/Priority:	High

Functional Requirement	Explanation
Requirement number:	4.30



Requirement name (use case name):	Search Message
Requirement short description:	The system must allow the tutor to be able to search for messages they have already sent to or received from students.
Requirement detailed description and constraints:	• The system allows the tutor to search for messages that they have sent or messages that they have received from a particular student or group messages. • The system must allow the tutor to search for certain key words or based upon filtering options or search criteria such as Date and time of the message being sent or the sender's/recipient's name. • The system will perform the search based upon user provided search input/criteria, simultaneously validating the search input by the user and then display the messages to the tutor in a desired fashion if the validation was successful.
Business rules applicable to this requirement	• The user must be a tutor in order for them to search messages addressed to them or sent by them. • The user must be logged into the system.
Revision date and Revision number:	2021-05-04 Version 1.0
Criticality/Priority:	Low

Functional Requirement	Explanation
Requirement number:	4.31
Requirement name (use case name):	Maintain Message
Requirement short description:	The system must allow the tutor to delete a message that they have sent or received.
Requirement detailed description and constraints:	• The system must allow the tutor to delete a message that they have sent to a specific student/group if they wish to do so. • After a search has been performed by the system based on specific search criteria provided by the tutor, the messages are displayed in a list format and allows the tutor to delete the messages in the list.



	• If a message is not searched for, the tutor must still have the option to delete a message even though the tutor did not search for a specific one.
Business rules applicable to this requirement	• The user must be a tutor in order for them to delete messages addressed to them or sent by them. • The tutor must be logged into the system. • A message cannot be deleted after 30 minutes have elapsed after the message has initially been sent.
Revision date and Revision number:	2021-05-04 Version 1.0
Criticality/Priority:	Medium



6.2.4 Payment Sub-system

Functional Requirement	Explanation
Requirement number:	5.1
Requirement name (use case name):	Search Payment
Requirement short description:	The system must allow the user to search for a specific payment log made by a student.
Requirement detailed description and constraints:	The user must be able to navigate to the payment content of the proposed Learn2Code system via the navigation bar. The user should then be able to enter the students first name in a search bar. The system will then return a list of all the payment logs made by that specific student that match the first name entered in the search bars. Constraint(s) The user may provide faulty search criteria which will produce incorrect results.
Business rules applicable to this requirement	• A payment and payment log has to be made before it can be searched. • The user has to be logged in.
Revision date and Revision number:	2015-02-19 Version 1.0
Criticality/Priority:	Medium

Functional Requirement	Explanation
Requirement number:	5.2
Requirement name (use case name):	Generate Invoice
Requirement short description:	The system should be able to generate an invoice for the following month that can be sent to the student automatically.



Requirement detailed description and constraints:	When the 25 th day of the month occurs the system automatically generates an invoice for the following month. An invoice contains information about the payment such as: The date, type of payment (either for a single course or a package), amount in question and students details. Once completed it is sent to the student. Constraint(s): None
Business rules applicable to this requirement	• Invoices must be stored on the Learn2Code system database.
Revision date and Revision number:	2021-02-19 Version 1.0
Criticality/Priority:	High

Functional Requirement	Explanation
Requirement number:	5.3
Requirement name (use case name):	Generate Payment Log
Requirement short description:	The system should be able to retrieve information from Paystack about a transaction and summarise it.
Requirement detailed description and constraints:	The system will retrieve and populate the following fields in a form from Paystack automatically: the payment amount, the student who made the payment, the type of payment and the day it was paid. Once completed the log is saved to the system. Constraint(s): None
Business rules applicable to this requirement	Payment logs must be stored on the Learn2Code system database.
Revision date and Revision number:	2021-02-19 Version 1.0
Criticality/Priority:	High



6.2.6 Report Sub-system

Functional Requirement	Explanation
Requirement number:	6.1
Requirement name (use case name):	Generate tutor details report
Requirement short description:	The system must allow a user to generate a report containing a list of tutors and the modules that they teach at TDO.
Requirement detailed description and constraints:	The system must allow an administrator to click on a button that will enable them to generate, view and/or download a report containing the name, surname and email of every tutor that is on the system along side with the modules that they tutor to make important managerial decisions such as to restructure the whole tutor hierarchy and re assign tutors to different or new modules added to the system that TDO will offer.
Business rules applicable to this requirement	
Revision date and Revision number:	2021-04-09 Version 1.0
Criticality/Priority:	Low

Functional Requirement	Explanation
Requirement number:	6.2
Requirement name (use case name):	Generate Sales Report
Requirement short description:	The system must automatically generate a sales report at the end of each month displaying relevant sales data to the administrators
Requirement detailed description and constraints:	- The system must generate a report at the end of every month automatically containing details of sales made during that particular month. - The report must contain the name and surname of the student who bought a package, the number of Standard and Pro packages purchased, the date of every transaction and the total sales in Rand value.



Business rules applicable to this requirement	• Only administrators must be able to have access, view and download the sales report. • No human interaction is allowed to generate this report, only the system can generate this report. • The sales report must be downloadable for managerial purposes. • A user with admin roles must be logged into the system to be able to view and download this report.
Revision date and Revision number:	2021-04-09 Version 1.0
Criticality/Priority:	High

Functional Requirement	Explanation
Requirement number:	6.3
Requirement name (use case name):	Generate attendance report
Requirement short description:	The system must allow a user to generate and view the amount of students who registered for a session and compare those results to the amount of students who actually attended the session.
Requirement detailed description and constraints:	• The user must be able to click a button to generate a report containing the name and surname of the students who registered and the total of students who registered for a session and compare it against the names and surnames of students who actually attended the live session with the total amount of students who actually attended the session.
Business rules applicable to this requirement	• Only users with administrative and tutor roles are allowed to have access to the attendance report on the system. • The user must be logged in to be able to view the attendance report. • The report must be downloadable on user request.
Revision date and Revision number:	2021-04-09 Version 1.0
Criticality/Priority:	High



Functional Requirement	Explanation
Requirement number:	6.4
Requirement name (use case name):	Generate Feedback Report
Requirement short description:	The system must automatically generate a report that contains feedback received from students based on their experience of the tutor's performance
Requirement detailed description and constraints:	- On the last day of the month, the system will generate a report containing all relevant information with regards to feedback received from students throughout the month. - The report will contain a bar chart with the tutors on the x-axis and the number of stars that each tutor has received in total from the students will be on the y-axis. This will allow management to see which tutors has the least stars to enable them to make decisions with regards to if the tutor must be questioned, trained or replaced by another tutor. - Below the bar chart will be a table for each tutor and all comments made by students for that particular tutor. The amount of tables will be equal to the amount of tutors who received some comments and recommendations. - The report will be displayed to the administrator and be downloadable on request.
Business rules applicable to this requirement	- Only users with administrative roles are allowed to have access to the feedback report. - The admin must be logged in to be able to view the feedback report. - The report must be downloadable on request by the admin. - If there is no feedback from any students during the month, the report must not be generated.
Revision date and Revision number:	2021-05-04 Version 3.0
Criticality/Priority:	Medium

Functional Requirement	Explanation
Requirement number:	6.5
Requirement name (use case name):	Generate total students report



Requirement short description:	The system must allow an administrator to generate a report on total students who currently owns a package at TDO with relevant details to the students.
Requirement detailed description and constraints:	• The system must allow an administrator to generate a report on all of the students currently on the database on request. • The report contain a few different tables explained down below. • The first table shows the students from a certain university. • The second table contains details about which year at university the students are currently enrolled for. • The third table shows all students that does the Pro package. • The forth table shows all students that does the standard package.
Business rules applicable to this requirement	• Only users with administrative roles are allowed to generate this report. • The admin must be logged in to the system to make use of this functionality.
Revision date and Revision number:	2021-04-09 Version 1.0
Criticality/Priority:	Medium

Functional Requirement	Explanation
Requirement number:	6.6
Requirement name (use case name):	Generate tutor's total session report
Requirement short description:	The system must allow an administrator to generate a report containing a list of all tutors and the amount of sessions each tutor has done for the month.
Requirement detailed description and constraints:	• The system must allow an administrator to click on a button that generates a report containing a list of all the tutors in the database with their name and surname. The list must also display how many sessions each individual tutor has presented for the month. The amount of sessions that a tutor does will be done by the system with an integer increment. When ever a session ends, the tutor's number of sessions conducted for the month will increase by one.
Business rules applicable to this requirement	• Only users with administrative roles are allowed to generate this report. • The admin must be logged in to the system to make use of this functionality.



	The report must be downloadable on admin request
Revision date and Revision number:	2021-04-09 Version 1.0
Criticality/Priority:	High

6.3 Non-Functional Requirements list

Requirement:	Explanation:
Requirement number:	1.
Requirement title:	Response time/efficiency
Requirement short description:	The system must allow for quick response time.
Requirement detailed description and constraints:	- The system will have a quick response time and fast throughput due to performance tuning and reduced data redundancy in the database. - The systems response time should only be affected by external factors like hardware faults that are out of the system designers' control
Requirement type:	Non-functional
Revision date and Revision number:	2021-04-01 Version 1.0
Criticality/Priority:	High



Requirement:	Explanation:
Requirement number:	2.
Requirement title:	User friendliness
Requirement short description:	The system will be easily navigated and updated. There should be clear instructions for users on what input is needed from them
Requirement detailed description and constraints:	• The system will be easy to navigate due to a descriptive navigation bar. • The system will be easily updated by the organization (TutorDevOps). • New features must be easy to implement. • User manuals should be available for users of the system
Requirement type:	Non-functional
Revision date and Revision number:	2021-04-01 Version 1.0
Criticality/Priority:	High

Requirement:	Explanation:
Requirement number:	3.
Requirement title:	Compatibility/flexibility



Requirement short description:	The system should be flexible to use on multiple types of devices and be customizable
Requirement detailed description and constraints:	• The system must be compatible with IOS, android, desktop and the web. This contributes to the systems flexibility. • The system should also be customizable for the directors to add new universities, degrees and modules from the different IT fields
Requirement type:	Non-functional
Revision date and Revision number:	2021-04-01 Version 1.0
Criticality/Priority:	High

Requirement:	Explanation:
Requirement number:	4.
Requirement title:	Security
Requirement short description:	The system should have security measures in place for access control and data protection purposes. Student information needs to be kept save.
Requirement detailed description and constraints:	• After 30 minutes of inactivity the user will be logged out and prompted to login to the system again. • Only authorized users(admin users) may access student data on the system. The system allows for different levels of control over certain data



	All user information must be protected (by use of the data protection act).
Requirement type:	Non-functional
Revision date and Revision number:	2021-04-01 Version 1.0
Criticality/Priority:	High

Requirement:	Explanation:
Requirement number:	5.
Requirement title:	Reliability
Requirement short description:	The systems accessibility is relied upon by its users
Requirement detailed description and constraints:	- Web and mobile versions of the system must be accessible during all times by any possible users. - Data should be stored on a secondary location as a back-up - The system should be reliable enough to not crash during use
Requirement type:	Non-functional
Revision date and Revision number:	2021-04-01 Version 1.0



Criticality/Priority:	High
-----------------------	------

Requirement:	Explanation:
Requirement number:	6.
Requirement title:	Reduced cost
Requirement short description:	The system will allow for reduced costs
Requirement detailed description and constraints:	• Because of the system integrating different applications it causes a reduction in the costs for the system owners as certain subscriptions can be stopped. • It will have the same affect for students using the system as they also do not need to incur these costs anymore. • The system should only have a minimal cost of hosting the website as well as server costs for data storage purposes.
Requirement type:	Non-functional
Revision date and Revision number:	2021-04-01 Version 1.0
Criticality/Priority:	High

Conclusion

From the above section it is seen that the system has a total of 92 functional requirements and 6 non-functional requirements. The functional requirements are split into 6 sub-systems and descriptions and details are given on each functional requirement in table format.



7. Feasibility/Decision Analysis

Introduction:

The feasibility /decision analysis has a main objective of identifying potential substitutes or alternatives to the new proposed system. The proposed system (Developed by Installing Inspiration team) will be compared to two alternative solutions. Throughout this section we will discuss the purpose of the feasibility study briefly, followed by a background statement of our client's "As-Is" system and requirements. Thereafter we will discuss the scope and the structure of the feasibility matrix. We will then draw up the feasibility analysis matrix discussing each aspect of the matrix and giving thorough recommendations and a conclusion.

7.1 Purpose, Background, Scope and Structure:

7.1.1 Purpose:

The purpose of the feasibility study is to identify and compare two other possible solutions to our client's current business needs and problems. The feasibility study is divided into four main sub studies defined as follows:

- Operational Feasibility: This sub study focuses on how beneficial a solution will be regarded to an organization and to what degree and extent the solution will address the organization's business requirements and business problems.
- Technical Feasibility: The solution will be graded according to the technical expertise it provides for the client as well as needed resources to fully install, maintain and operate the solution successfully.
- Economic Feasibility: The solution will be analysed to determine if the economic benefits brought forth from the implementation of the solution will outweigh the total costs to the client. Payback periods, subscriptions, installation, and support fees with calculations will be used to provide further information towards an informed conclusion.
- Schedule Feasibility: This will determine how long it will take to complete the project/solution for the client to use combined with a project timeline as well as time available for the project.

7.1.2 Background

TutorDevOps currently runs their online tutoring business by the combination of VOIP software (Discord to host their online classes/sessions) , social media platforms (WhatsApp for real-time communication with regards to when live sessions will be held and business updates and payment processes to be followed) and other third party communications channels such as Gmail to communicate with their current and prospective customers (students) and their website where students can choose and purchase certain packages with their own benefits and merits.

TutorDevOps specifically focuses on the education and tutoring of Coding languages, how they are relevant to certain industries, current coding trends and practices as well as practical usage and expansion of their customer's knowledge of the software developing industry. Currently TutorDevOps are seeing a dramatic increase in prospective students in need of tutoring, with specific attention to coding, across a few universities in South Africa, and realized the need for a system that integrates their "as-is" system into their own, personally tailored, independent tutor management software. TutorDevOps also identified their dependence on a wide variety of different third-party software (Gmail, Discord, Google meet, GitHub etc). TutorDevOps requires a system that integrates all these technologies into their own "To-Be" system to ultimately create a user friendly, dynamically adjustable solution, that caters for their valued customers all at one place, the TutorDevOps home.

7.1.3 Scope and Structure:

The feasibility study will compare the following two solutions to the proposed solution (Learn2Code): "Tutor Cruncher" and "Teach Works" software respectively. Because a business's needs changes depending on their current financial position, business life cycle, competitors and other external and internal constraints, opportunities or weaknesses, a business will regard certain sub studies of the feasibility matrix more important than others. We therefor decided to weigh the four sub studies accordingly:

Operational: 40%. The client's main problem is with regards to software currently used that are not affiliated with each other and no compatibility is provided between these software programs. The client is also currently aiming at expanding their system usability to their customers/students and making their services and packages available all at one place. This is the highest priority and need to the client.

Schedule: 25%. The client is seeing an increase in customers as the coding and software industry is rapidly becoming one of the most sought-after categories for potential job opportunities given the steep increase in information processing needs in all industries. Information is regarded as one of the most valuable assets in the current day global economy, there for a new system is needed to



cater for their customers to be as user friendly as possible. The client needs a competitive advantage/edge over their competitors as the competition in their industry is growing rapidly by the day. This should be done as soon as possible to obtain and maintain that advantage before someone else seizes this opportunity before they can.

According to an article by Hafele software, The US Bureau of Labour Statistics projects that software developer jobs will increase by 30% between 2016 and 2026. They are of the opinion that South Africa will easily see the same trend but are concerned that South Africa will “struggle to fit those positions if the universities do not keep up, and if schools across the country do not introduce coding earlier, to ensure school-leavers are as equally marketable as those in other countries”. This provides just another incentive for TutorDevOps to usher in their needed system as soon as possible to surpass any potential new threats (other companies with the same target market as TutorDevOps) to enter the market they are doing business in (See references at the end of the section on Feasibility/Decision analysis)

Economic: 20%. The main objective of the client is to produce a sustainable revenue over the course of the next few years in relation to time spent tutoring their customers/students. Given that TutorDevOps is the client owner’s secondary business (or business on the side), their main source of income is acquired by their full-time job as software developers and system analyst. The TutorDevOps business they established together was a dream and brought forth by their passion for coding and selected hobby. We therefor weighed the economic feasibility as third priority.

Technical: 15%. The client owners are very experienced in their field working with hardware and software, so the technical feasibility is not as important, given their field of expertise and the service they provide to customers/students. The tutors they hire to work at TutorDevOps are required to have superior knowledge in coding languages and applications and it’s implied that the tutors, who are coders themselves, will be able to adapt and use the system to the best of their abilities without a lot of training. Training is not necessary when it comes to coders, who are tech savvy, to be able to use the system properly. The client has the appropriate software and hardware infrastructure to carry the processing, storing and operational capabilities that the software requires to run effectively and efficiently.



7.2 Feasibility Analysis

7.2.1 Feasibility Analysis Matrix

Feasibility Criteria	Weight	Candidate 1  Learn2Code	Candidate 2  TutorCruncher	Candidate 3  TeachWorks
Operational Feasibility Functionality. A description of to what degree the candidate would benefit the organization and how well the system would work. Political. A description of how well received this solution would be from both user management, user, and organization perspective.	40%	- The Learn2Code system is specifically tailored and designed for the TutorDevOps operations and functionalities fully meeting functional and operational requirements. Learn2Code is not a software built as a general tutoring management system that any business can use and tweak or change to their liking, its sole purpose is to identify the business problems of TutorDevOps solely and address them accordingly to every specific detail. - Learn2Code solution will include a fully functional web app and a mobile application designed to connect tutors and students together everywhere they might go. Covid-19 forced the world to move to a more online approach with regards to education, making Learn2Code the best means of learning and teaching simultaneously, bridging external constraints with ease.	- TutorCruncher is a software developed by the TutorCruncher team whose focus is on handling administrative functions for tutoring businesses and has been used by multiple tutoring agencies across South Africa for the past few years. - TutorCruncher provides features such as business analytics to analyse a client's performance with graphs and revenue breakdowns, Customer relationship management by use of emailing and SMS mediums of communication, Intelligent web design with over 15 years of experience in relation thereto, Scheduling of classes, payment management and other online teaching tools. - TutorCruncher is a desktop-based application but it can be integrated with a client's	- TeachWorks is a simple, easy-to-use system for managing teachers, students, schedules, billing, and other features as a solution for tutor management and administration intensive tasks. - TeachWorks offers a variety of features to meet a general tutoring company's needs and requirements. - TeachWorks adds a scheduling feature to co-ordinate schedules between multiple teachers, students, and location differences. This is an appropriate solution to TDO's scheduling needs but given the fact



		• Learn2Code allows potential tutors to apply for a tutoring position at TutorDevOps (here after referred as TDO) and concurrently allowing the business owners to manage, approve and reject tutor applications on the web app itself. • The system will cater for the creation, addition and deletion of any modules the TDO management sees fit, fulfilling the administrative role of manually creating and managing modules. • The solution includes packages (subscriptions on a monthly/ semester basis) that students can select from with regards to specific modules, making the options to the students more attractive and make the students feel free to choose the best package that best suits their needs. • The packages offered are: ○ Standard (Two 1–2-hour group sessions per week) ○ Pro (Two 1–2-hour sessions per week which will be a one-on-one with a tutor) • The solution is dynamic in the sense that the different types of modules can be added according to different universities. • Currently TDO is using WhatsApp to communicate and schedule session times with students. The solution,	already existing website. This will require an integration effort by the TutorCruncher support team and TDO owners at the expense of an additional cost and time efforts. • TutorCruncher has a functionality that automatically matches tutors with students, but this proposition is not in alignment with TDO's business rules concerning allocations of tutors. TDO aims at personally being invested into their customers or students and therefor assigns one and only one tutor to a specific module to have interpersonal and professional relationships with their students, ultimately establishing great tutor-student bonds to keep current students loyal to TDO, rendering this functionality from TutorCruncher basically void. • TutorCruncher offers 47 different personalized emails that are triggered by a host of events and time spans. TutorCruncher offers SMS as a medium of communicating lesson reminders, interview invitations and invoice notifications. The technology used by	that TDO mainly focuses on conducting business and live sessions online, location is not a problem or a requirement that needs fulfilment. There also need not to be a feature to co-ordinate scheduling between multiple tutors/teachers considering that TDO's employee models in relation to students and classes are set up in a fashion that only one tutor will be assigned to a course to give their undivided attention to those students who specifically need to build their knowledge in that specific module. This results in TDO not needing to use these features but still pay for them none the less, being an unnecessary cost to cover. • The calendar function provided by TeachWorks
--	--	--	---	---



		proposed by Learn2Code, is by implementing a calendar functionality on the web application where tutors can add slots and open a poll where students can be able to vote for a session which fits them best (The polling functionality is specifically designed for group sessions, the standard package). Communications with tutors can be done via the web applications or the mobile application, with a full-on real-time communications network, making it easier and faster to communicate with and between students and tutors, taking the place of WhatsApp and making TDO somewhat independent. This also forces the students to continually interact with the mobile application specifically, always keeping them up to date with communications, Scheduling with regards to live sessions, reminders and other relevant communications. • Student, tutor and admin profiles will be managed by the system with an option to alter personal profile content. • Secure accessibility to the system will be provided for with allocated roles for tutors, students and administrators. • Reminders before every live session will be communicated to students	TutorCruncher with regards to communication mediums is somewhat outdated as SMSs are being replaced with real-time, fast and internet based interactive platforms such as WhatsApp for example. • TutorCruncher adds a very helpful tool to allow students to give valuable feedback to their tutors/ tutoring business on how their experience was with their service and allow the client admins to monitor tutor effectiveness based on user-received feedback. • TutorCruncher adds a feature that allows tutors to share documents with individual students or a group of students all at once. • TutorCruncher provided for a mobile app for the tutoring agency and staff, but this mobile app is not accessible to students for communication on all platforms. • TutorCruncher has broad capabilities such as video calls and collaborative workspaces where student and tutor can simultaneously work on the same document. This is an appealing feature, but the tutoring system	allows students and teachers to filter the calendar content by monthly and daily agenda's which will not meet TDO's needed solution to be as flexible and dynamic as possible when it comes to the timing and scheduling of live sessions and classes. • Teacher availability is a useful feature added that is available to students to track and determine when a tutor will be available on the schedule, however this option be offered by TDO due to the fact that students and tutors communicate directly with each other to find a slot suitable for both parties to attend the live sessions. • Personal Calendars are a beneficial add-on to the required system to allow students who
--	--	--	---	--



		to ensure that they attend the meeting and get their value out of the sessions. • Learn2Code will also cater for resources, study materials, recorded sessions and notes all at one place on the system. • The solution will include a reporting sub-system functionality to help the TDO owners to make informed business decisions and strategies. • A basket checkout functionality is also provided by the solution increasing the user-friendliness to students dramatically. • If a student does not want to have regular and live tutoring but would rather study content on their own time, the Learn2Code solution provides an alternative to packages (subscriptions) called courses: • Providing students to pay for a course with included videos, tutorials, step-by-step learning and study content which a student can complete on their own time and in their own way. • The courses also have no limit or expiration date making the courses a very attractive alternative, but it also simultaneously allows tutors to create short videos for the courses and no time spent on individual sessions!	is a broadly based system to fit into many different subjects and industries of tutoring. A collaborative workspace might be effective for a math's class, but this will seem pointless for coding as the approach to learning coding is much different to how a tutor would approach a math class. Adding to this, a coding class or lesson will most likely be conducted in a coding platform or software such as visual studio code or visual studio community edition, where screen sharing would suffice, and collaborative tools will cause a possible disruption in coding practices and demonstration. • TutorCruncher adds tutor-student matching by location, by skill or by customer category. Given that TDO's business model is differently implemented than the TutorCruncher is providing for, this feature will not be used by the TDO team. • For example, TDO is strictly online based, meaning that tutor-student matching by location will not be used at all. All tutors are approved by the TDO team themselves to	selected the pro subscription package to track their individual sessions with the tutor for the module they bought the package for. • Students and tutors can create their own personal, secure profile on the website to gain access to the content of the system and use it to their benefit. Accounts can be updated and deleted accordingly. • Automatic email reminders are sent to students to improve attendance to the tutoring sessions. • Automatic SMS reminders are sent to students as well about lessons taking place. SMS is an outdated means of communication in the constantly changing technology and social environments. • A billing and payroll feature is
--	--	--	--	--



			potentially approve and assign a tutor to a specific module based on their attributed skills. Score: 80/100	also provided for to manage employee salaries/wages and customer billing accounts. TDO currently has an external accountant keeping track of all billing information, payment receipts, income statements and other relevant financial details. The billing and accounting feature will not be utilized by TDO. Score: 85/100
		Score: 95/100		
Technical Feasibility Technology. An assessment of the maturity, availability (or ability to acquire), and desirability of the computer technology needed to support this candidate.	15%	- The Learn2Code system can be hosted and managed by the client's current infrastructure with regards to the web API, database management tool and the mobile application. - The proposed solution supports a mobile and computer-based approach using technologies such as Angular, My SQL Management tool, Microsoft visual studio community edition, Ionic and Electron.	- This solution is developed for windows/Linux only. There are web integration options available if the client so wishes to continue with it. - This is not a web based or cloud-based approach making the software more IT resource intensive (GPU, RAM and storage intensive). TDO's current infrastructure will be sufficient to operate	Teachworks offers free integration and add-ons in addition to the software. Teachworks accounts can be integrated with leading web services and software. Add-ons can be added to core features to cater for specific business needs of the client.



Expertise. An assessment to the technical expertise needed to develop, operate, and maintain the candidate system.	• Learn2Code will be fully planned, designed, developed, tested and implemented by the Installing Inspiration team. The team has all the capabilities and understanding of the technology to properly deliver a working and effective system for TDO. • No additional time by the TDO owners or management is necessary in the development of the final solution except requirement gathering meetings applicable to defining the scope of the system properly. • The client can hire between 1-3 people with appropriate knowledge to always handle and manage the system. Learn2Code is flexible in the sense that more administrators can be added to the system the bigger the system grows. The system grows with the business and no upgrades in terms of software requirements are needed to cater for an increasing customer/tutor/administrat or base, simply add a new role! • Training with regards to the system will be very minimalistic and the TDO owners and current tutors can easily train a new admin, tutor or student to familiarize themselves with	and effectively run the software easily. • TutorCruncher has an API called “TutorCruncher API” that the client can use to access all information in the TutorCruncher account. This allows the client to get an overview of large amounts of information or it can also be used to edit specific parts of the system quickly. (Refer to Appendix B). Technical knowledge and background of API’s and structure are needed to be able to use this feature to its purpose. • The mobile app is available to admins only and requires some experience and knowledge to be able to use. • TutorCruncher has highly skilled software developers who can train and educate TDO on how to use TutorCruncher to the full capacity. • Explanations and guides for tutors and users are not readily available for new users to the system.	• Integrations currently offered: ○ Stripe ○ QuickBooks ○ Microsoft Excel ○ MailChimp ○ Lessonspace ○ Zapier ❖ Add-ons catered for by Teachworks: ○ Website booking plugin. ○ Website profiles ○ SMS reminders ○ Teachworks API (requires additional training to be understand the system and use if effectively) ○ Join open classes. ○ Lesson Requests ○ Custom Enrolment forms ○ Invoice autopilot ○ Remote login ○ And many more ❖ All add-on featured requires
--	---	--	---



		how the system works. Fast, simplistic, flexible and effective. Learn2Code adds a feedback option where students can share their commendments and concerns with their tutors and system administrators to constantly give constructive feedback to the TDO team to gradually improve the system to a near desired perfection in terms of user friendliness. Score: 75/100	The usage of this system requires a lot of knowledge on the basic processes to be followed when using the system and the user guide is not extensive enough, therefor training will be needed and even explanation videos to students as well to have the best experience possible. Score: 70/100	minimalistic training to use. ❖ Teachworks is a web-based solution which means that there are no installations required to use the software. ❖ Access to the software can be accessed anywhere in the world with internet access. ❖ No modifications to current hardware, software and network infrastructure is needed by TDO, except each tutor or student needs to have a laptop or mobile device to use the software features. Score: 80/100
Economic Feasibility Cost to develop:	20%	❖ The solution will be done by the Installing Inspiration team from design and planning up unto final delivery at no cost at all. ❖ Since the delivery of all milestones of the solution will be conducted online, there are no printing costs associated with the system.	❖ TutorCruncher is an already existing application for desktop and mobile which implies that there is no cost to develop the system. ❖ TutorCruncher offers a two-week free trial to their application/software .	❖ Teachworks is a web-based solution implying no installation fees or costs to develop the solution for the client. ❖ Software integrations offered by the solution is free of charge.



Monthly subscriptions :		❖ No training fees will be charged, no monthly subscription or support fees will be charged, or upgrade/maintenance fees are associated with the proposed solution.	❖ TutorCrunch offers three packages mentioned in the discussion points below: ❖ In-app chat support is an additional R185pm accruing to $(185 \times 12) = R2220$ a year ❖ Phone and video call support is an additional R1850pm accruing to $(1850 \times 12) = R22\ 200$ ❖ Pay as you go (Monthly): A base Fee of R565 covering licensing and basic support + 1% of revenue earned for the month. Included are 3 months of free chat support and all platform features and unlimited users and lessons.	❖ Add-ons will also not cost anything to implement. ❖ Teachworks offer a free three-week (21 days) trial to use their software to decide whether or not you want to subscribe or use their system for your business as a solution. After the three weeks have lapsed you must pay to use on a monthly basis from that point onward. ❖ Teachworks offer the following three monthly subscriptions (All packages offer unlimited students, teachers, user accounts, emails, add-ons): ❖ Starter: \$15 per month plus \$0.25 per student lesson monthly. This option excludes API.
---------------------------------------	--	---	--	--



			Yearly costs to subscription will be $(R565 + [0.01 \times [60 \times 400]]) \times 12 = R9\ 660$ $R9\ 660 + (R185\text{pm} \times 9 \text{ months}) + R22\ 200$ =R33 525 annually cost to business ❖ Startup (Monthly): A base Fee of R1350 covering licensing and basic support + 0.65% of revenue earned for the month. Included are 3 months of free phone support and all platform features and unlimited users and lessons. Included in this package is also free training. Yearly costs to subscription will be $(R1350 + [0.01 \times [60 \times 400]]) \times 12 = R19\ 080$ $R19\ 080 + (R1850\text{pm} \times 9) + R2\ 220 = R37\ 950$ ❖ Enterprise package: This package's pricing will depend on the circumstances of the client's business. TDO is currently not under this bracket as	\$1 = R14.78 (Currency exchange valued on 31 March 2021) TDO has +- 60 students $\\$0.25 \times 14.78 = R3.70$ $\\$15 \times 14.78 = R221.70$ $([R221.7] + [R3.70 \times 60]) \times 12 = R5\ 324.4$ annually ❖ Growth: \$45 per month plus \$0.10 per student lesson monthly. This option includes Teachworks API Access. $\\$0.1 \times 14.78 = R1.48$ $\\$45 \times 14.78 = R665$ $([R665] + [1.48 \times 60]) \times 12 =$ R9 045.60 annually ❖ Premium: \$175 per month plus \$0.05 per student lesson monthly. This option adds API and the first 5 branch accounts for free.
--	--	--	--	--



			the client (TDO) must have a revenue or R10 000 000 annually. This option will then be disregarded from this point on onwards. When TDO uses the “start-up” option they will receive a fairly decent ROI but, given that TDO is still a very young and financially small business, they have a lot more risk associated with picking this option, as the cashflow is relatedly low annually, creating a risky investment scheme with better investment alternatives at hand.	$\\$0.05 \times 14.78 = R0.74$ $\\$175 \times 14.78 = R2\ 586.5$ $([R2\ 586.5] + [R0.74 \times 60]) \times 12 = R31\ 570.80$ annually TDO will not use the starter option given that the API is a very helpful tool and that the TDO team is extremely experienced with API usage for years now, giving them a managerial edge over competitors. The premium package proves to be quite expensive, but a very helpful addition is the branch accounts to be used for future expansion. Score: 75/100
		Score: 100/100	Score: 70/100	
Schedule Feasibility An assessment of how long the solution will take to design and implement.	25%	❖ The total timeline for the project from planning to implementation and full system completion and delivery will take 7-9 months. Breakdown follows: ❖ Requirement elicitation: 2-4 weeks ❖ System proposal: 3 weeks	❖ This solution is already designed and existent but customizing the software to TDO’s desires will take 1 month to fully alter and use effectively. Additional training of 1-2 weeks will be necessary with	1. Teachworks is a web-based solution therefor there are no development needed as the system is already developed and readily available. 2. This solution is a generic tutoring



		❖ System functional specification: 1 Month ❖ Prototyping, technical design and test specifications for Learn2Code: 4-6 weeks ❖ System testing and documentation: 2 months. ❖ Fully tested system + user documentation: 4-7 weeks ❖ Final deliverable and project handover will be done on the 10th of November 2021.	applicable fees related to the latter. ❖ TDO's website must be integrated with the desktop application which will take an additional month with consequential integration fees and even more training needed between 1-2 weeks	software which means that the client must rebrand the system to their liking, which will take approximately 1 month to completely rebrand to the client's liking. 3. Software training will take about 3 weeks for tutors and administrators 4. Software training for students will take a week to get completely used to. Score: 75/100
		Score: 65/100	Score: 75/100	
Ranking:	100 %	85.5%	75.25%	79.80%

7.2.2 Recommendations:

The ranking of the Three solutions are as follows (Based on total weighted averages):

1. Learn2Code (Developed by the Installing Inspiration team) – 85.50%
2. Teachworks – 79.80%
3. TutorCruncher – 75.25%

1. Learn2Code

The Learn2Code proposed system is specifically designed to bridge the gap between the “as-is” and the “to-be” system and to overcome barriers identified in the system requirements elicitation. This solution will be designed with close collaboration with the TDO business owners throughout the development of the solution to find any missed or newly identified requirements the final system must include or have. Technical feasibility will be of little concern as the TDO owners are very



confident in their coding abilities, application of knowledge with respect to the “to-be” system and common understanding of the business and system model proposed and requested. Given that the proposed solution is not so hard-and-software-intensive, the current IT infrastructure owned by TDO is more than enough to run the Learn2Code solution smoothly. Any upgrades to the system can be compensated with an increase in hardware capabilities such as RAM allocation, external or internal SSD upgrades and replacement as well as regular backup control and management to sensitive data. There are no costs associated with the development to the business giving this proposal a very high advantageous outlook for the TDO team. 7-9 Months are needed to successfully develop the final system, but the other benefits to why this solution would be the perfect fit, significantly outweighs the timing constraint (which is not a high priority to TDO).

2. TeachWorks

This solution is a well-designed and established software but proves to be quite expensive to operate on an annual basis. This is a risky investment but used correctly with the right market conditions will result in a prominent outcome. The solution unfortunately caters for a lot of unnecessary features that will be paid for but not used by the system users and administrators. Some of the features offered does not meet the requirements of the most important need for TDO which is a dynamically adaptable scheduling system on short notices and provides no features for online class integration with the software. The scheduling aspect of the solution is a very attractive, but it is worth the wait to wait for a system that is designed to meet all TDO’s and only TDO’s system requirements. No rebranding is applicable with no intervention from the Installing Inspiration team.

3. TutorCruncher

TutorCruncher in terms of operational feasibility has a decent number of features to generally manage the TDO’s administrative intense tasks to do, but the system also adds numerous features that are of no value to TDO, causing them to pay a lot more for what they will use. Some of the features offered by this solution does not even fit into the business model that TDO currently operates and will operate in with regards to the “to-be” system, rendering quite a few features of no value. There are quite a few costs and fees such as phone and call support, integration fees and licensing fees as well as relevant training fees associated with it. Training will be necessary for tutors and students to use the system to the best of their own interest. The mobile app added as an extension of platform usability is only available to TDO administrators and the owners, therefor leaving a massive opportunity unutilized to integrate the students and tutors to get together on different platforms, ultimately not realizing a crucial need identified by TDO to get the communication between the tutors and students closer and more flexible (mobile or web-based



options). The supportive documentation such as administrator, tutor and student user manuals are a big key advantage added to the system, with other supporting documentation and manuals on how to fully integrate the TutorCruncher API with a client's website. The design of this proposal is well suited and user friendly, but unfortunately there are very important system requirements that are not met by this alternative.

Conclusion:

Based upon research done on the two alternative candidates and our proposed system, the individual sub studies of the feasibility study matrix, weightings and ranking, it is clear to see that the Learn2Code solution will best fit our client's current need and fill in the problems faced by the business. In terms of operating feasibility, the highest priority sub study, there is compelling incentive to implement the Learn2Code system fully dedicated to the TutorDevOps business needs and user requirements (Tutor, Admin and student based). In terms of economic feasibility there are absolutely no costs associated with the complete implementation of our proposed solution which is a substantial benefit to the TDO owners. The only potential shortcoming is found within the schedule feasibility but given that TDO is currently still small, and the owners are not dividing their full attention to TDO, due to TDO being their side-business and secondary source of income, this will not be proved to pose a problem for the foreseeable future. We as a team are solely dedicated to the development of their "To-Be" system which will give them a sense of security with regards to this potential and possibly only shortcoming. The benefits of choosing Learn2Code absolutely outweigh the potential disadvantages/obstacles.



References:

Osarome.blogspot.com. 2021. *1. TECHNICAL FEASIBILITY 2. OPERATIONAL FEASIBILITY 3. ECONOMIC FEASIBILITY*. [online] Available at: <<https://osarome.blogspot.com/2011/10/1-technical-feasibility-2-operational.html>> [Accessed 30 March 2021].

Haefele Software | Outsourced .NET Development Specialists. 2021. *Software Development Continues To Be The Most In-Demand Job In SA*. [online] Available at: <<https://www.haefelesoftware.com/software-development-continues-to-be-the-most-in-demand-job-in-south-africa/#:~:text=The%20US%20Bureau%20of%20Labor,percent%20between%202016%20and%202026.>>> [Accessed 30 March 2021].

Tutorcruncher.com. 2021. *TutorCruncher • Tutoring Management Software*. [online] Available at: <<https://tutorcruncher.com/>> [Accessed 30 March 2021].

Help.tutorcruncher.com. 2021. *TutorCruncher Help Center*. [online] Available at: <<https://help.tutorcruncher.com/en/>> [Accessed 30 March 2021].

2021. [online] Available at: <<https://tutorcruncher.com/api/#list-all-action-types>> [Accessed 31 March 2021].

Teachworks.com. 2021. *Plans for Education Companies of All Sizes | Teachworks*. [online] Available at: <<https://teachworks.com/pricing>> [Accessed 31 March 2021].

Document Conclusion

In conclusion the aim of the project proposal was to identify the problems that the organisation TutorDevOps face and offer them a solution that would mitigate all their problems. Thus, the document consisted of several components which helped us analyse the organisation and identify solutions, an example of a component would be the preliminary investigation which allowed the team to identify the issues that the organisation faces and in turn the requirement analysis was used to understand what the capabilities the proposed system must have. The document is then completed with the client sign-off.



8. Appendix A: Client documentation

Introduction

The following section includes examples of documentation used by the client in their current way of doing business. It includes screenshots of their current website, invoices, emails, registration, and examples of adverts.

8.1 Documents, Forms and Manuals

8.1.1 Adverts

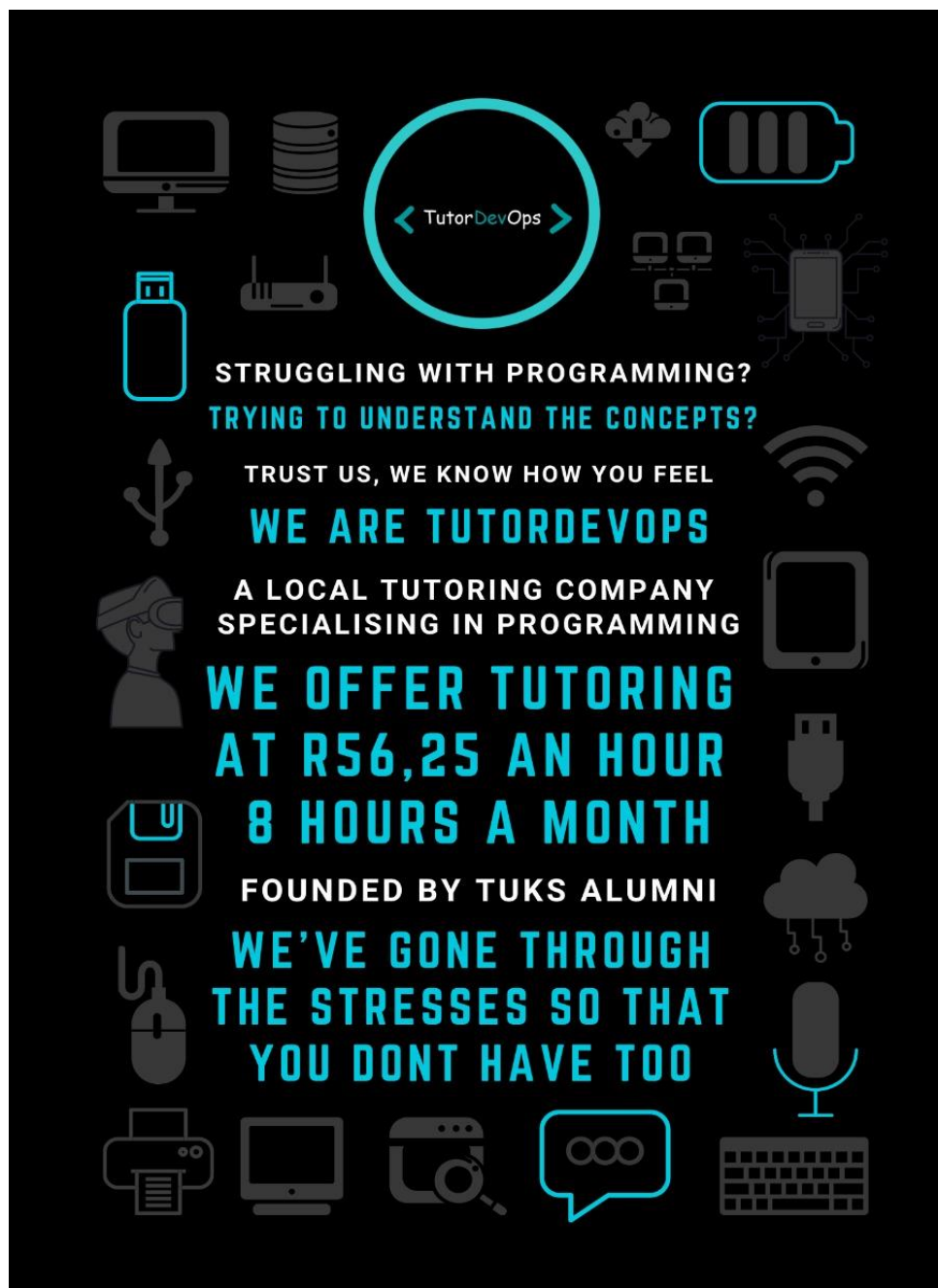


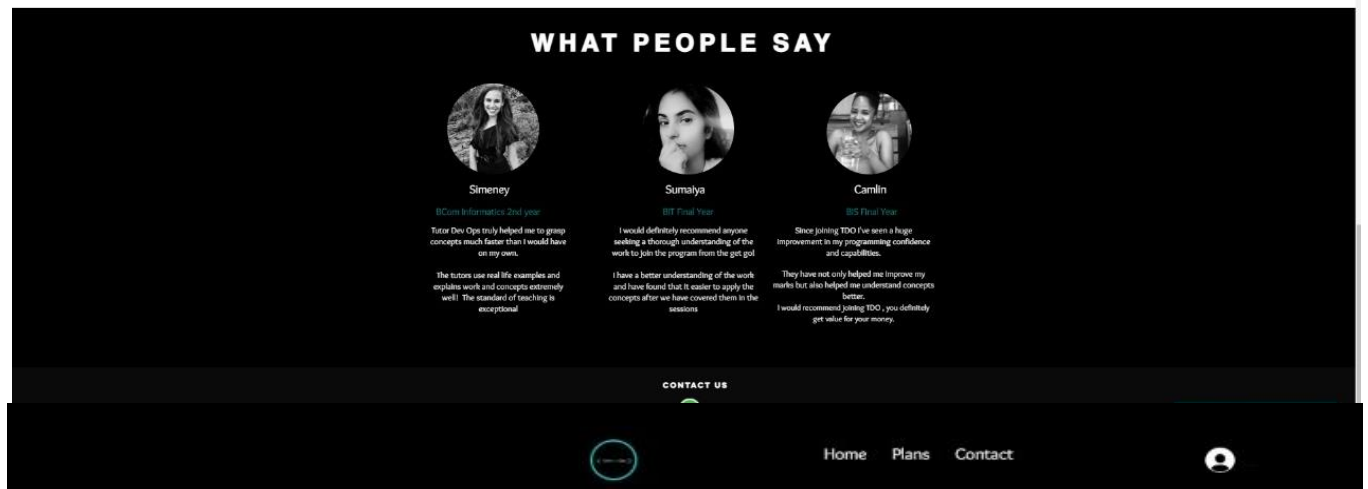
Figure 11 - TDO Advert



8.1.2 Current Website



Figure 12 - Mural 1 of TDO Website Screenshots



Tutor Plans

Noob Dev

R 450

valid for one month

Pay Now

Modules Covered

INF 154 - INF 272 - INF 214 - INF 354

COS 132 - COS 210

IFM 1

* Live recordings available for subscribed month only *

* Price covers 1 module only *

Noob Dev - Semester

R 1300

1st April - 30th June

valid for 3 months

Pay Now

Modules Covered

INF 154 - INF 272 - INF 214 - INF 354

COS 132 - COS 210

IFM 1

* Live recordings available for subscribed period *

* Price covers 1 module only *

Noob Dev Pro

R 1000

Includes Private 1 on 1 sessions weekly 1 hour per session

valid for one month

Pay Now

INF 154 - INF 272 - INF 354

COS 132

* Price covers 1 module only *

* Automatic access to Noob Dev perks *

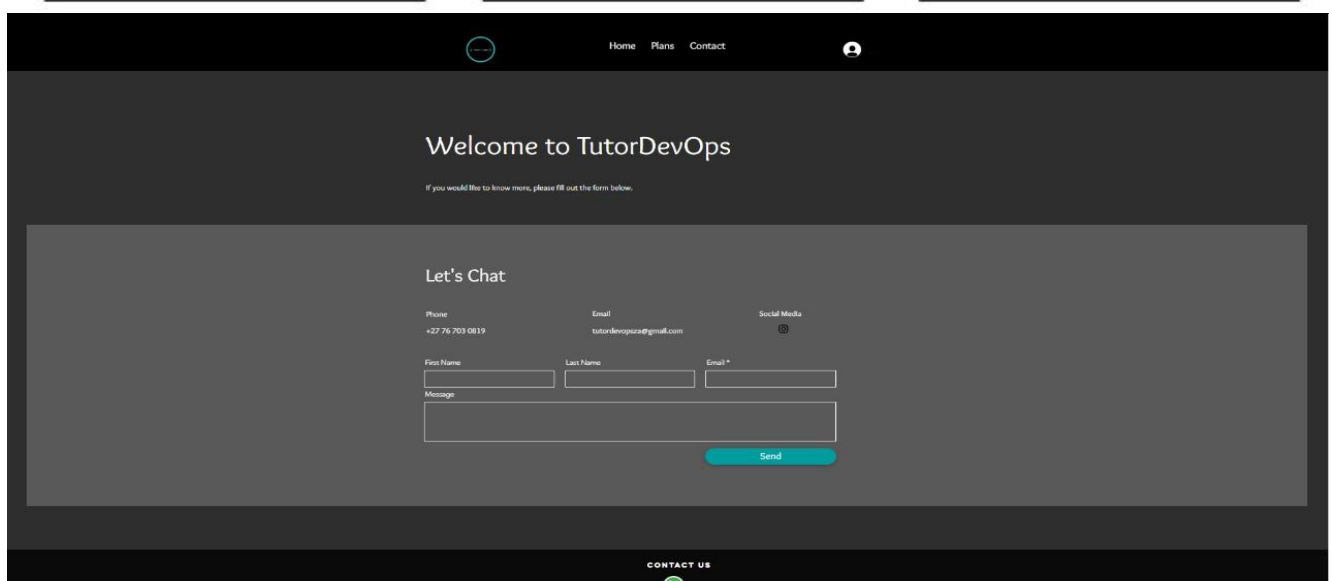


Figure 13 - Mural 2 of TDO Website Screenshot

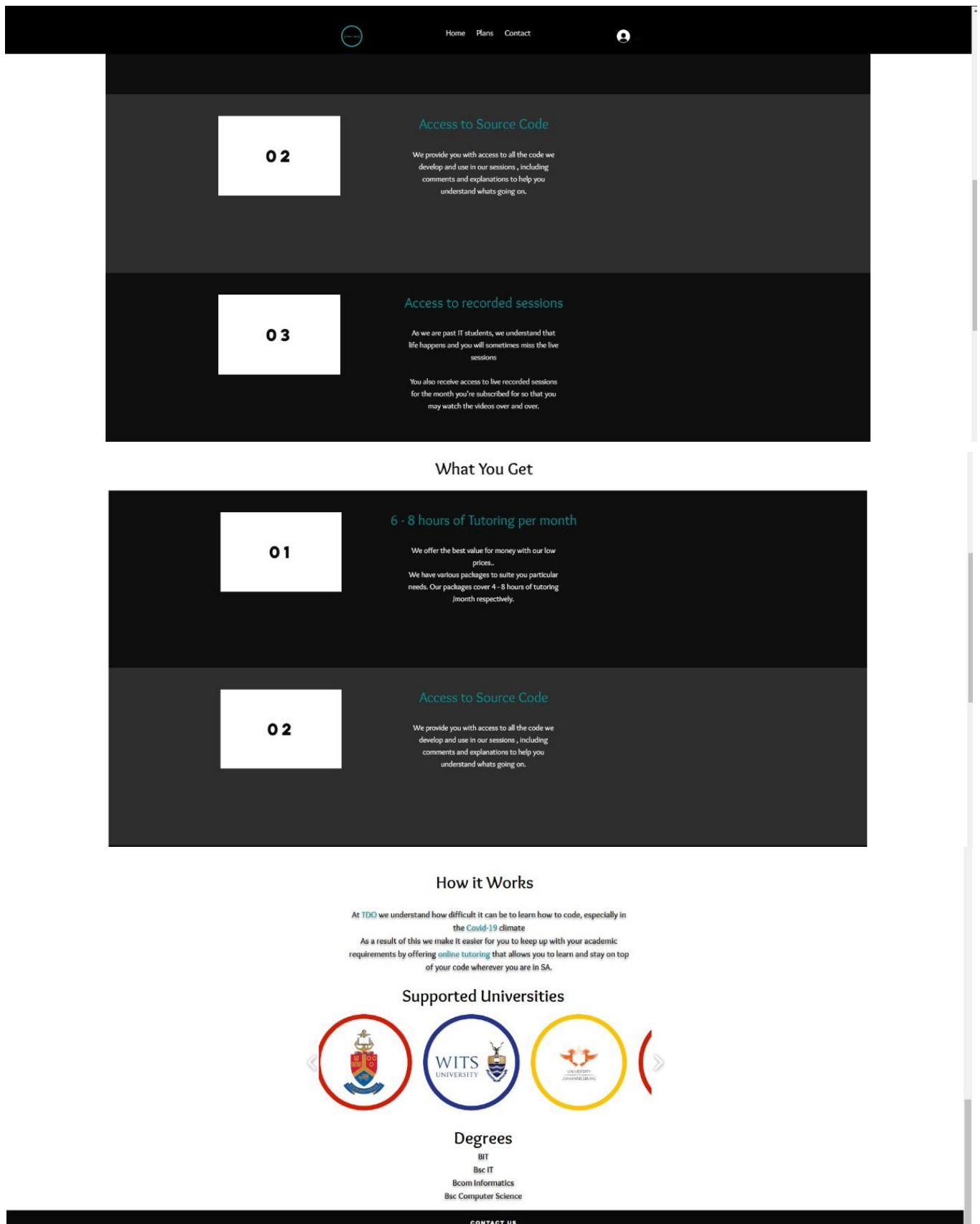


Figure 14 - Mural 3 of TDO Website Screenshot



8.1.3 Google Forms

Registration Form

Form description

Name *

Short answer text

Surname *

Short answer text

Degree *

Short answer text

What year are you in your current degree? *

☐ First

☐ Second

☐ Third

☐ Its complicated



What year are you in your current degree? *

☐ First

☐ Second

☐ Third

☐ Its complicated

Email *

Short answer text

What help are you looking for from TutorDevOps ?

Long answer text

Anything else you would like to add/ask ?

Long answer text

Figure 15 - Screenshot of TDO Google form



8.1.4 Emails

8.1.4.1 Welcome Email



Hi, welcome to TDO !!!!

Attached is the TutorDevOps Pdf with Information on Pricing and all other relevant Information. Below you will find all the relevant information required to kick start your programming.

This Email contains a few applications and/Online platforms that you will require in order to successfully become integrated with our tutoring style.

First Complete the registration Form below thereafter continue below

Click on the blue Link



Google Forms

[Registration Form](#)

Once Payment has been made you can continue below

All software mentioned below is Free and Open Source. Do Not Pay for anything

T-1:First We need to ensure that you are ready , please therefore **download and install** the following application **both on your laptop and cell**



[Discord App](#)

T-2:We need to ensure that you have an application installed that will allow us to **assist you remotely** on your laptop if you get stuck



[TeamViewer](#)

T-3:When coding we make use of an application called Github which will **make code available to you** for access **i.e Assignment solutions** or even **Practical suggestions in code format**. Create and Register an Account below ...



[GitHub](#)

T-4:After Successfully Installing the Applications Please Register A Discord Account and **Email Us Your**

- Discord User Name with the 4 digit code i.e : WarMachine#2453
- Your **Proof of Payment** attached
- Github Username ie : Warmachine

Once all these steps have been completed successfully TDV will add you to the relevant server platforms and will get in contact with you to discuss further proceedings

******Please remember to constantly check your emails******

If you have any other issues and queries please feel free to contact

TutorDevOpsZa on Whatsapp Here : **081 536 6864**

or alternatively email us at : tutordevopsza@gmail.com

One more thing Please join the Whatsapp group below

Figure 16 - Mural of Welcome Email

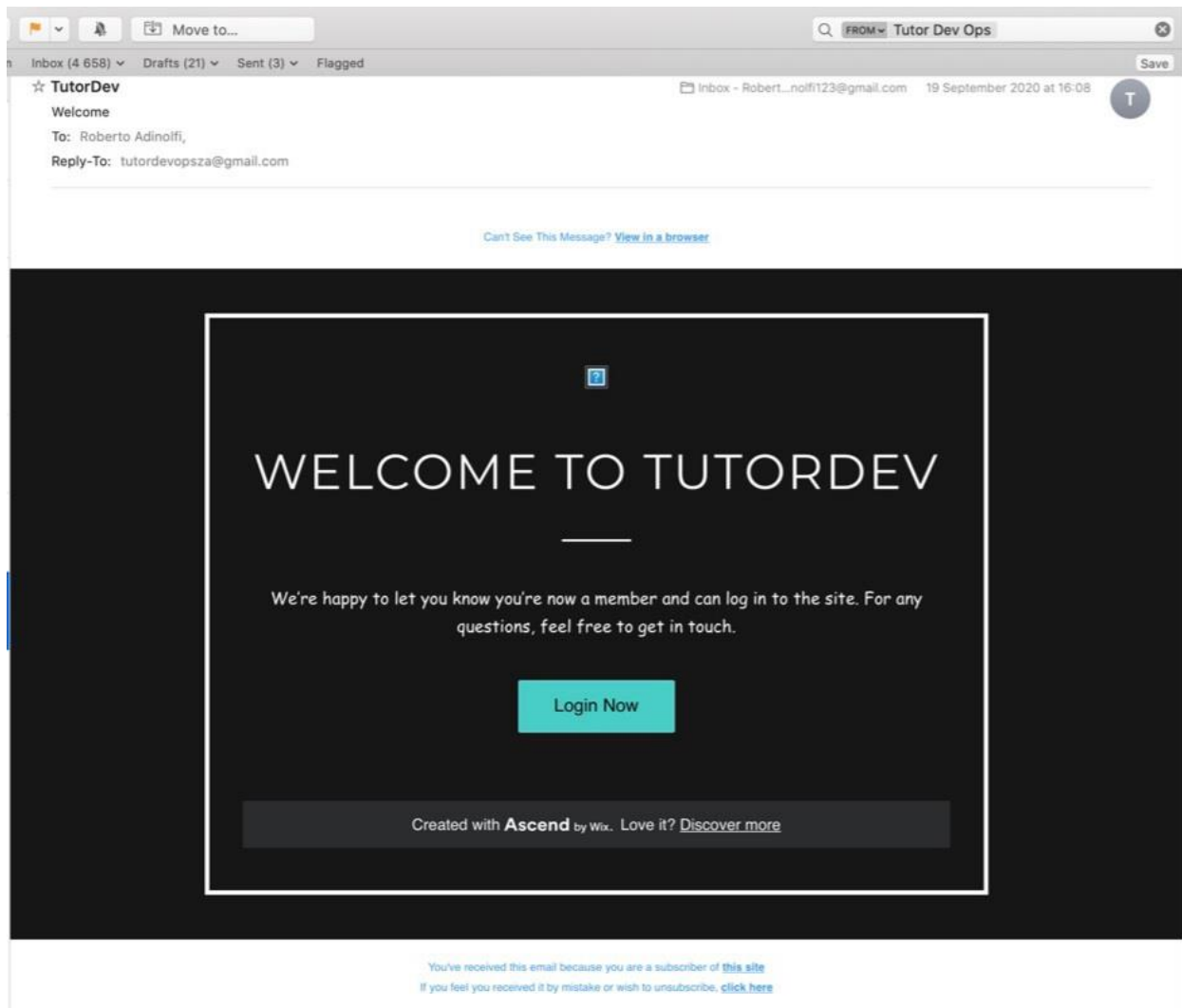


Figure 17- Screenshot of Welcome email



8.1.4.2 Notification Emails

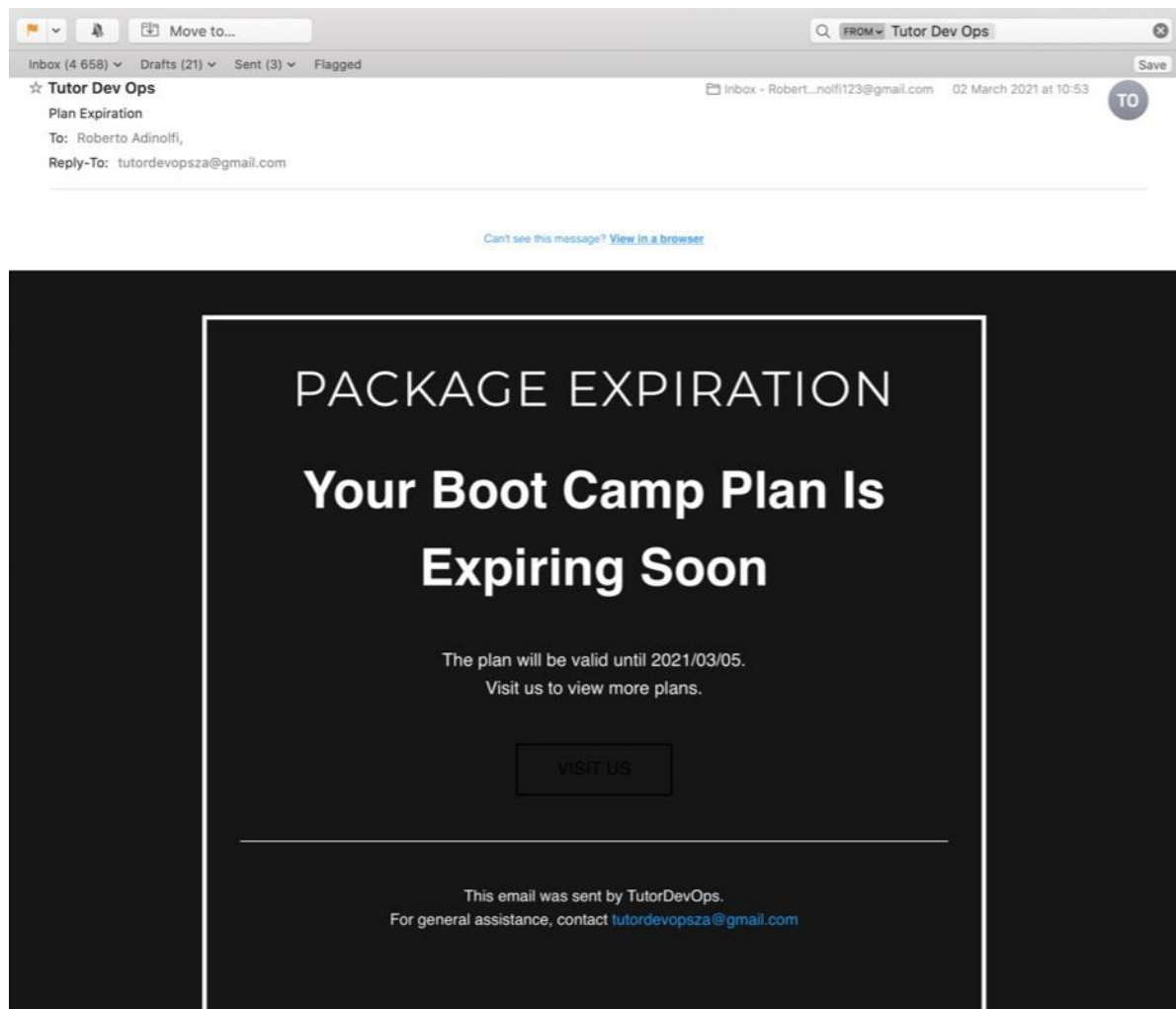


Figure 18 - TDO Notification Email



8.1.4.3 Order Confirmation Emails

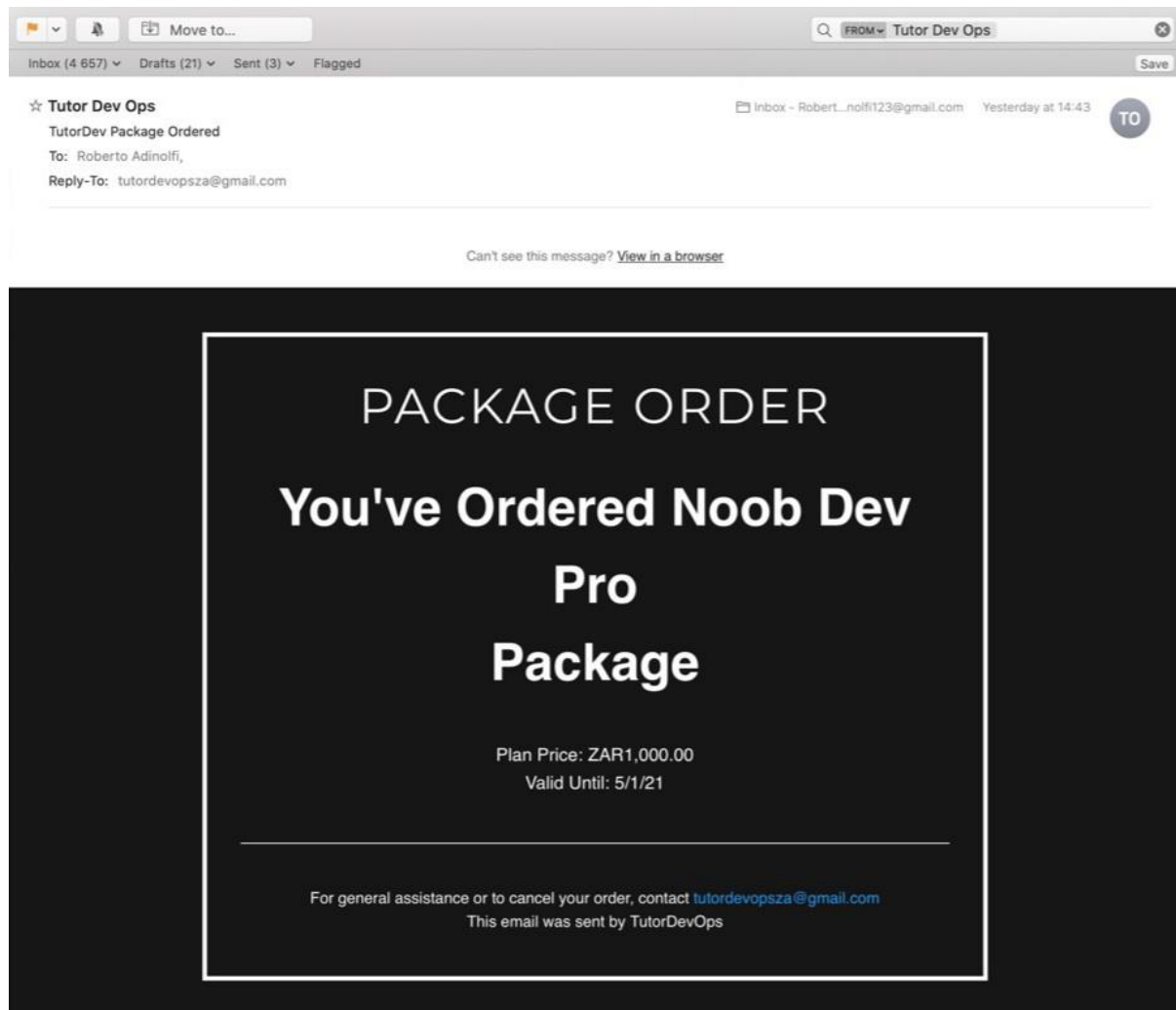


Figure 19 - TDO Order confirmation email



8.1.4.4 Package Expiring Emails

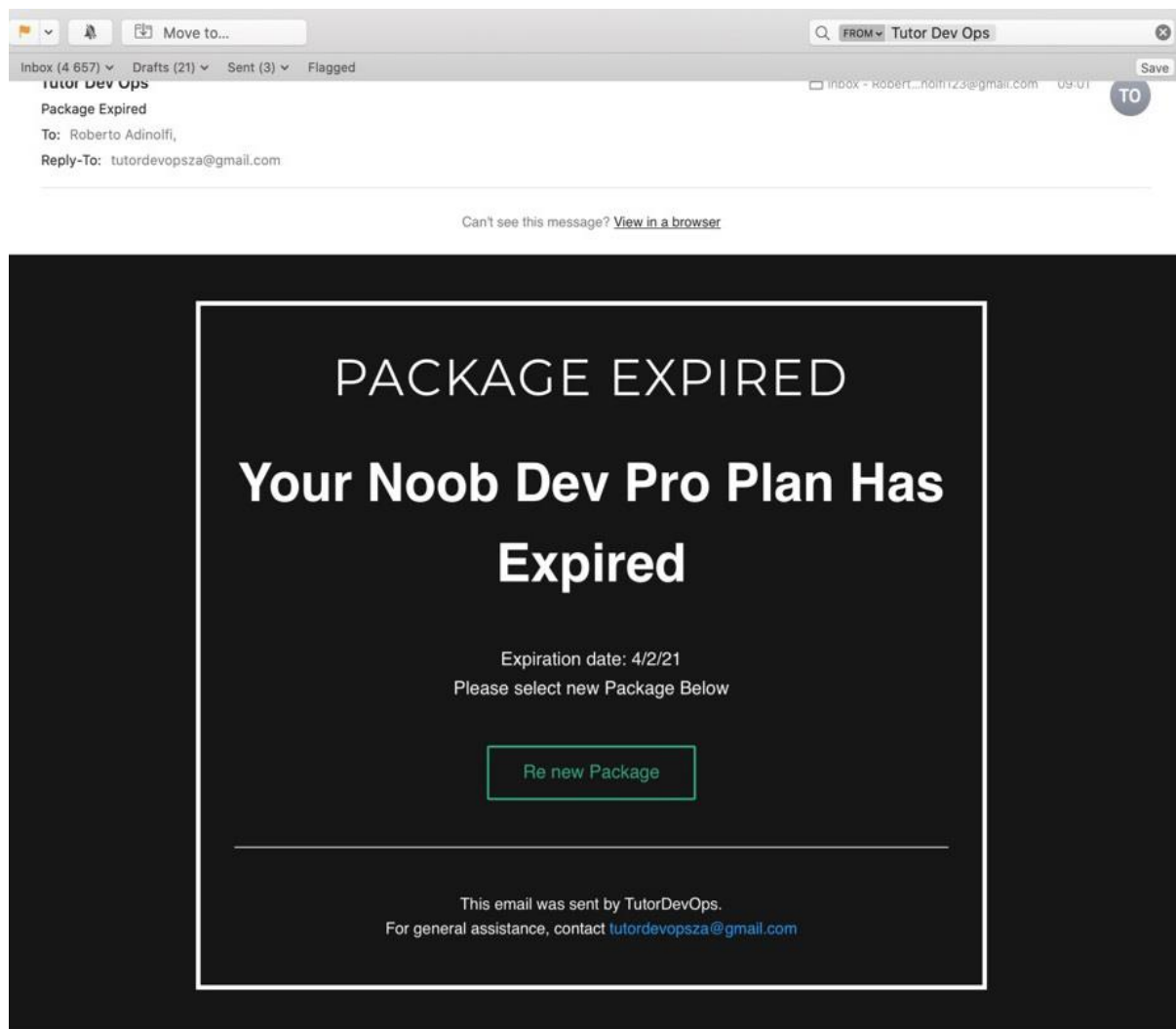


Figure 20 - TDO Package expiring email



8.1.4.5 WhatsApp Group

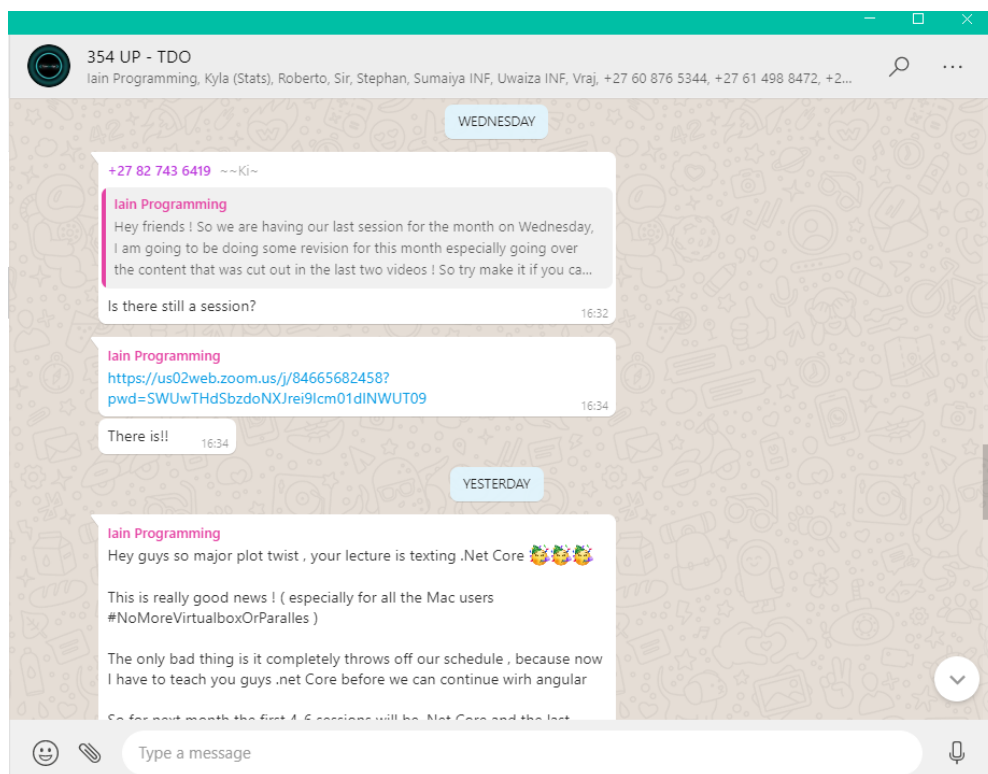


Figure 21 - TDO WhatsApp Screenshot 1

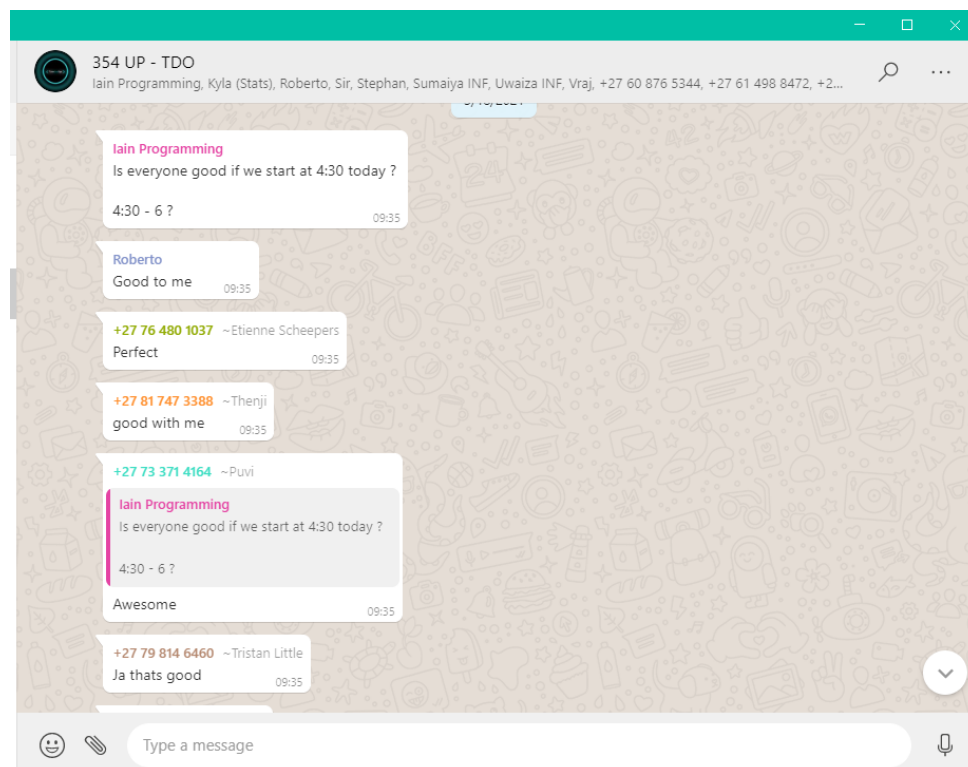


Figure 22 - TDO WhatsApp Screenshot 2

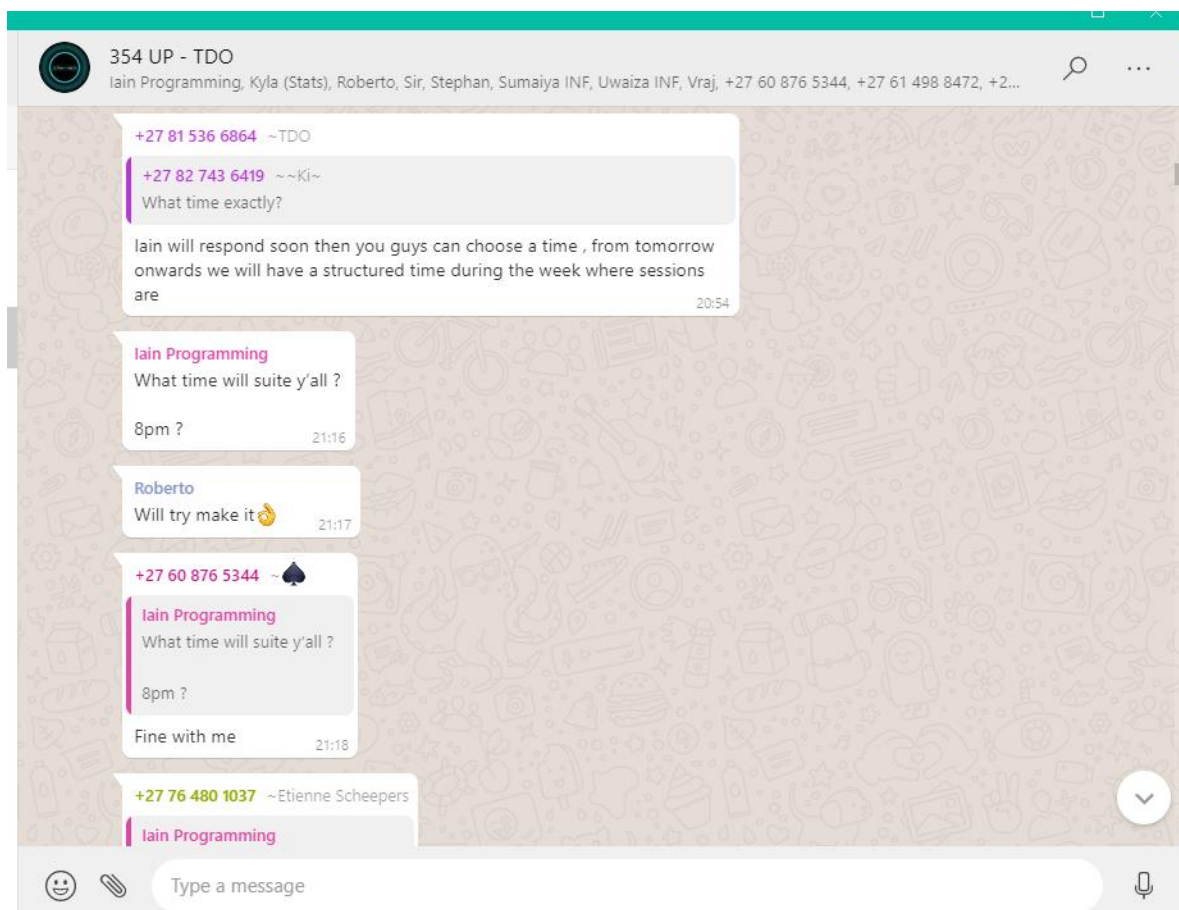


Figure 23 - TDO WhatsApp Screenshot 3



Figure 24 - TDO Invoice Example

From the above section TDO currently makes use of multiple communication platforms and that their current systems' functionality is limited. This confirms the inefficiency of the current system which is mentioned throughout the document.

9. Appendix B

Introduction

This section consists of a literature study and research conducted to find potential available systems to solve the business' problems and meet the needs of the client. A few examples will be given with screenshots and references as proof of the research done.

9.1 Literature Study

9.1.1 123Tutors

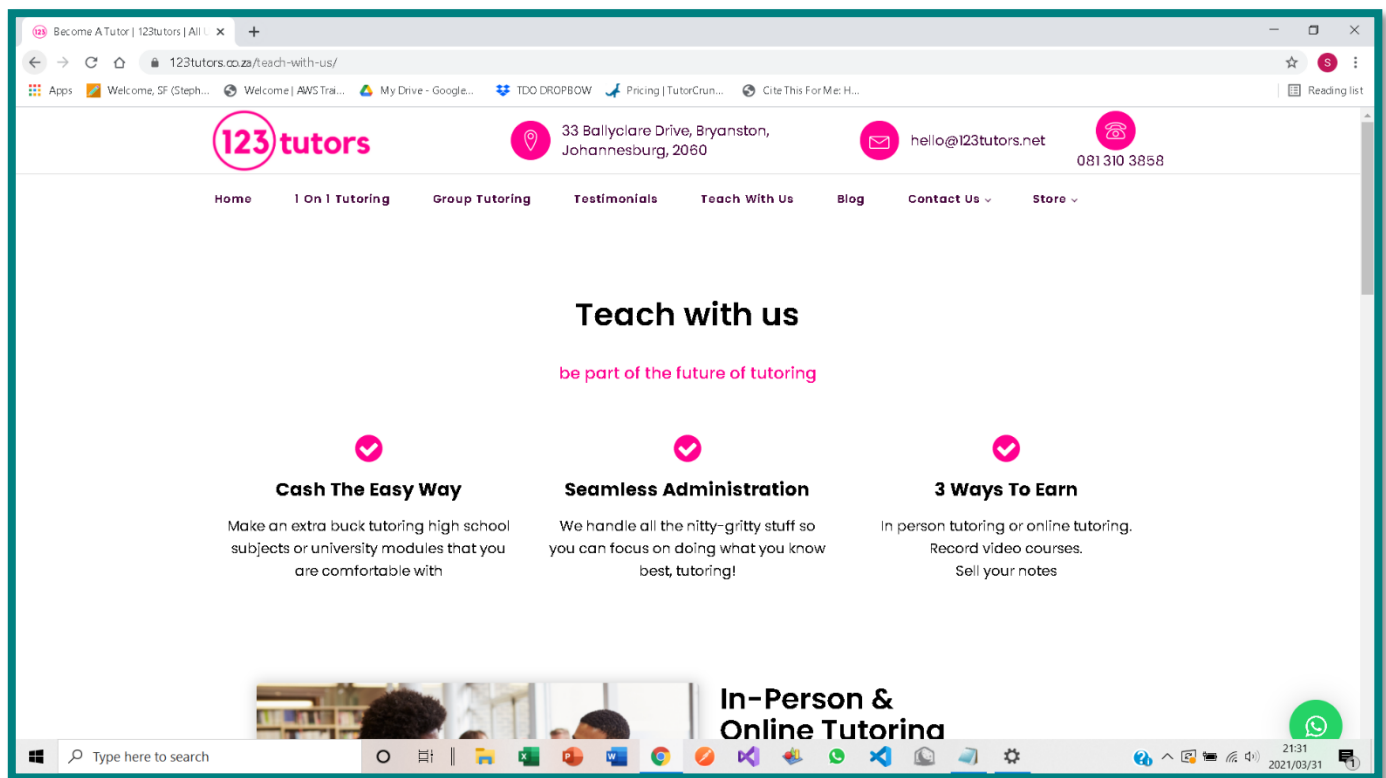


Figure 25 - 123Tutors Website Screenshot

123Tutors is an online company who mainly brings together tutors and students. We at first thought this would be a good solution to TDO, but when we did further research, we found out that they do not offer any software packages to be used by businesses as a managerial function. 123Tutors brings together any tutors with any students who go to their website to sign on. They themselves are the company who generates profits from this software as they only offer a place for students to find content and connect to a tutor and allow tutors to sign up on their website. 123Tutors offers 1 on 1 tutoring and group tutoring options and generates profits for the management thereof. This makes 123Tutors a competitor and not a potential solution to TDO's system problems. 123Tutors allows for tutors to apply to be tutors at 123Tutors.

9.1.2 TutorClass

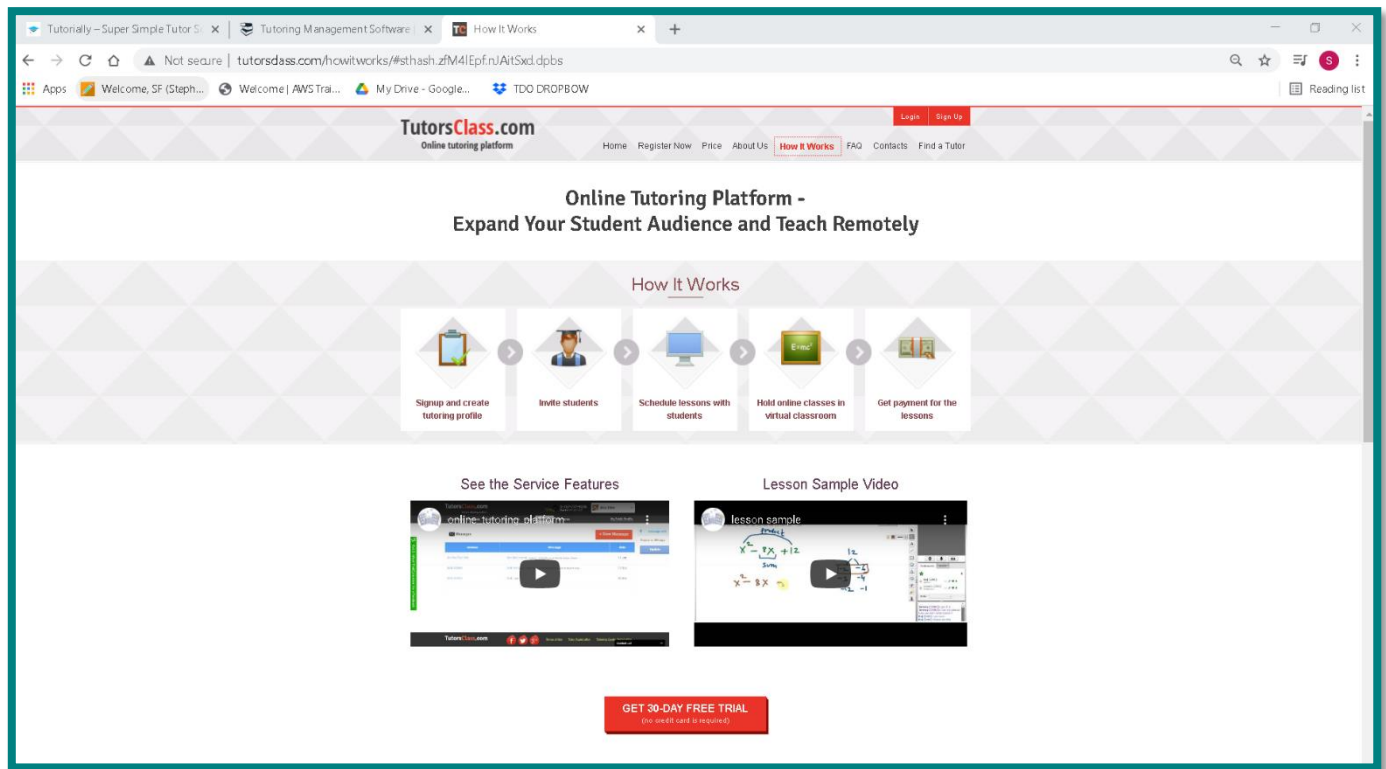


Figure 26 - TutorClass Website Screenshot

TutorClass is a special web-based software that completely substitutes offline tutoring. “You teach and we manage your business processes”. TutorClass platform allows you to teach anytime and anywhere, get free promotional webpage, get more students to find you via your promotional webpage and receive payments securely via PayPal. You can create your tutoring profile quickly and invite students to join you via your email. You can have pre-paid or post-paid methods of payment. Classes are held on a online classroom provided for by TutorClass. Track payments and messages from students to the tutor. We found that this software seems like a potential solution, but this software only accounts for the management of a single tutor’s payments, students, messages, and lessons + schedules. This software does not allow the client to have more than one tutor, as the software is specifically designed for one tutor to tutor students. TDO has multiple tutors and therefor this solution will not meet the essential requirements and fit into TDO’s business model and logic.

- Pricing:
 - Basic 12 months – \$0
 - Beginner - \$59.99
 - Advanced - \$299.99



9.1.3 TutorPanel

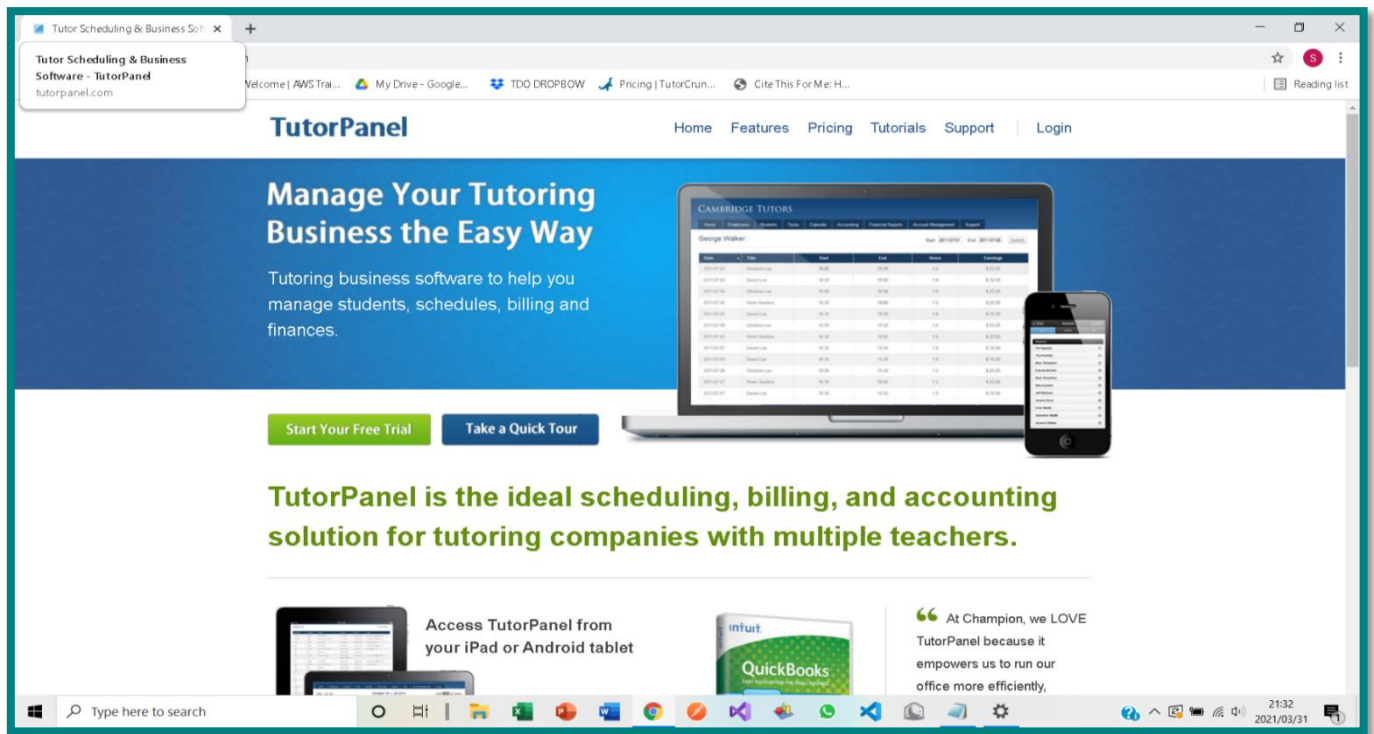


Figure 27 - TutorsPanel Website Screenshot 1

TutorPanel is a tutoring business software to help a client manage their students, schedules, billing, and finances. TutorPanel is heavily focused on finances which is not our client's priority at all.

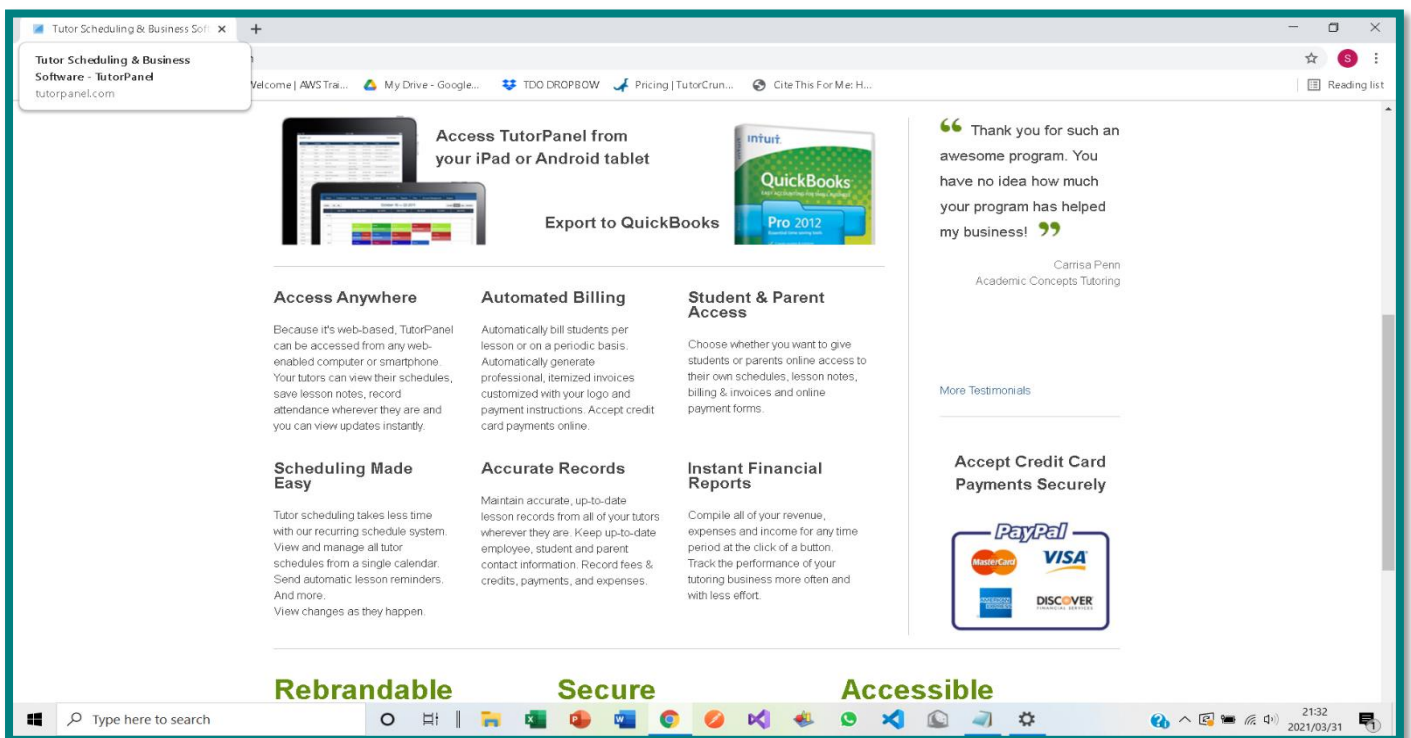
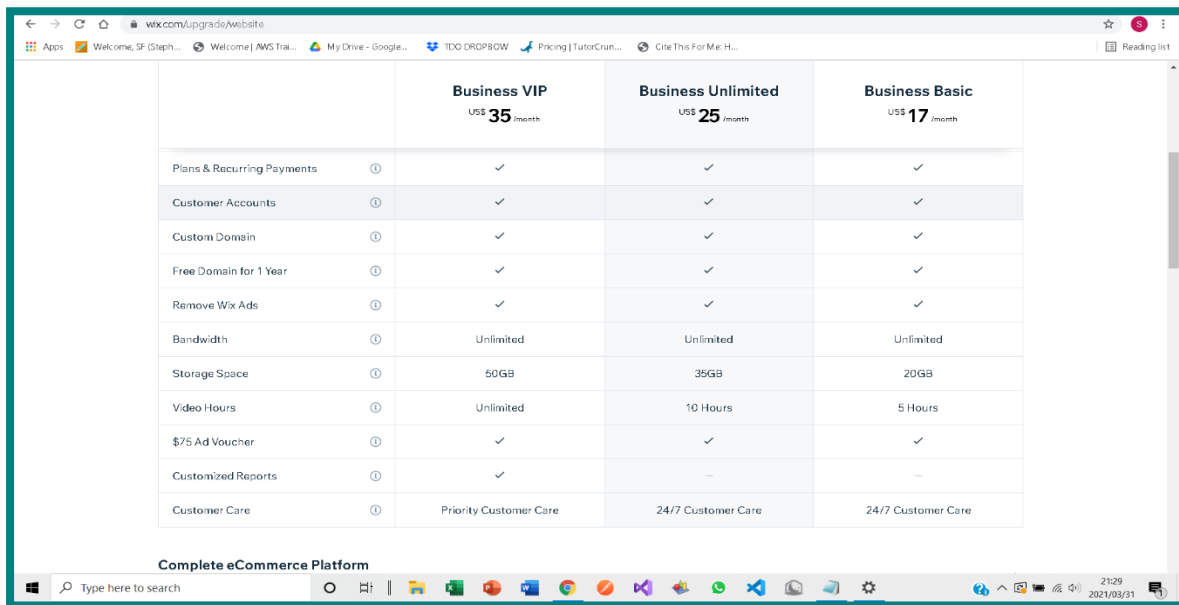


Figure 28 - TutorsPanel Website Screenshot 2

9.1.4 Wix

Wix.com Ltd. Is an Israeli company mainly focused on the development and distribution of software. Wix was found in 2006 by Israeli developers such as “Avishai Abrahami, Nadav Abrahami and Giora Kaplan. Wix is a cloud-based web service that provides for web development. Wix has offices in Brazil, Canada, Germany, India, Ireland, Lithuania, USA and Ukraine. Users can add plug-ins to their site, online marketing, e-mail marketing, e-commerce, contact forums and much more.



	Business VIP US\$ 35 /month	Business Unlimited US\$ 25 /month	Business Basic US\$ 17 /month
Plans & Recurring Payments	✓	✓	✓
Customer Accounts	✓	✓	✓
Custom Domain	✓	✓	✓
Free Domain for 1 Year	✓	✓	✓
Remove Wix Ads	✓	✓	✓
Bandwidth	Unlimited	Unlimited	Unlimited
Storage Space	50GB	35GB	20GB
Video Hours	Unlimited	10 Hours	5 Hours
\$75 Ad Voucher	✓	✓	✓
Customized Reports	✓	—	—
Customer Care	Priority Customer Care	24/7 Customer Care	24/7 Customer Care

Figure 29 - Wix Website Screenshot 1

The figure above displays the pricing to use Wix software and features. Wix offers Website Plans to showcase a professional site (Not TDO’s interest) and Business and eCommerce Plans which is mainly focused on accepting online payments. A list of pro’s and con’s will follow for the above mentioned figure as well as the figures to come (Wix snapshots). Pros and Cons are based on the current need of our client, TDO.

PROs	CONs
Customer Care	Free domain for only one year
Unlimited Bandwidth	Storage space is too little to handle the amount of video data added by TDO for lesson recordings
Wix adds removed which annoys people	
Video hours to stream videos online and allow client to offer subscriptions to content	



The screenshot shows the Wix website's 'Upgrade Website' page. At the top, three business plans are listed: Business VIP (US\$ 35/month), Business Unlimited (US\$ 25/month), and Business Basic (US\$ 17/month). Below this, a section titled 'Complete eCommerce Platform' describes the ability to 'Build, manage, and scale your business with Wix.' A table follows, comparing features across the three plans. The 'Business Unlimited' plan is highlighted as the 'MOST POPULAR' option.

	Business VIP US\$ 35 /month	Business Unlimited US\$ 25 /month	Business Basic US\$ 17 /month
Complete eCommerce Platform Build, manage, and scale your business with Wix.			
Unlimited Products	✓	✓	✓
Abandoned Cart Recovery	✓	✓	✓
Subscriptions	✓	✓	—
Multiple Currencies	✓	✓	—
Automated Sales Tax	500 Transactions/mo	100 Transactions/mo	—
Sell on Social Channels	✓	✓	✓
Sell on Marketplaces	✓	✓	—
Dropshipping by Modalyst	Unlimited Products	Up to 250 Products	—
Product Reviews by KudoBuzz	3,000 Reviews	1,000 Reviews	—
Loyalty Program by Smile.io	✓	—	—

Figure 30 - Wix Website Screenshot 2

PROs	CONs
Subscriptions	Unlimited products (TDO offers subscriptions and short courses, this feature is not going to be used)
Loyalty program to give loyal customers (students) some discount/ vouchers etc.	Automated sale tax (not needed by TDO)
Sell on social channels (marketing for TDO services)	Sell on marketspaces (TDO does not sell products rather services)

The screenshot shows the 'Wix Premium Plans' page. It features a header with navigation links (Creation, Business, Growth, Resources, Pricing, Support) and a 'Sign In' button. The main heading is 'Wix Premium Plans' with a subtext: 'Wix gives you 100s of templates, unlimited pages & top grade hosting FREE. Upgrade to Premium and get even more.' Below this is a 'Get Started Today' button. Two main categories are shown: 'Website Plans' (Great for showcasing a professional site) and 'Business & eCommerce Plans' (Essential for accepting online payments). A table at the bottom compares the three business plans, with 'Business Unlimited' marked as 'MOST POPULAR'.

	Business VIP Get the Full Suite US\$ 35 /month	Business Unlimited Grow Your Business US\$ 25 /month	Business Basic Accept Online Payments US\$ 17 /month
Secure Online Payments	✓	✓	✓
Plans & Recurring Payments	✓	✓	✓

Figure 31 - - Wix Website Screenshot 3

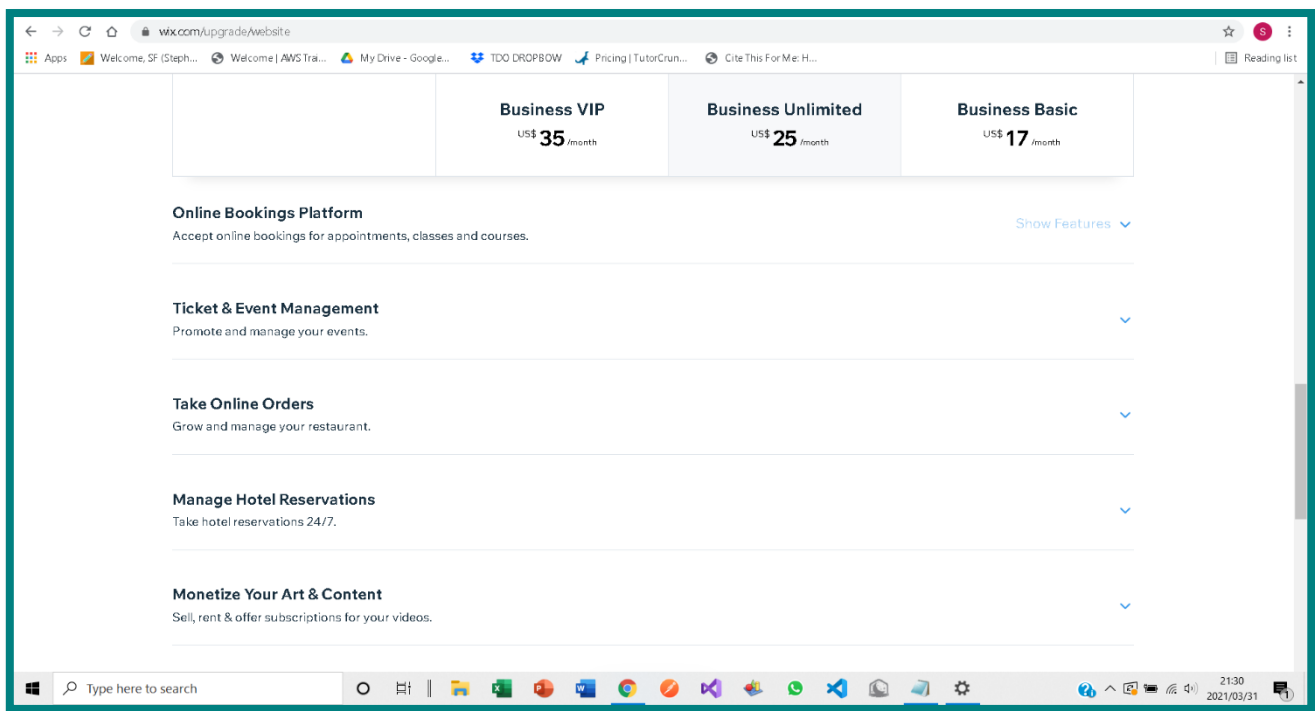


Figure 32 - - Wix Website Screenshot 4

PROs

Packages are relatively cheap to subscribe to.

CONs

Packages do not really cater for the TDO specifications.

- Online booking platform – Accept online bookings for appointments, classes and courses. This would be a feature that TDO will use regularly. Added to this feature is multi-calendar sync, Unlimited services and classes, Subscriptions and memberships and SMS notifications.
- Ticket and event management – TDO is strictly online based, leaving this feature void and unusable.
- Take online Orders – Grow and manage your restaurant (completely irrelevant to TDO's business)
- Manage Hotel reservations – Completely irrelevant to TDO's business.
- Monetize your Art and content – Completely irrelevant to TDO's business.

Subscription plans:

1. Business VIP - \$35 pm
2. Business Unlimited - \$25 pm
3. Business Basic - \$17 pm



9.1.5 Tutorially

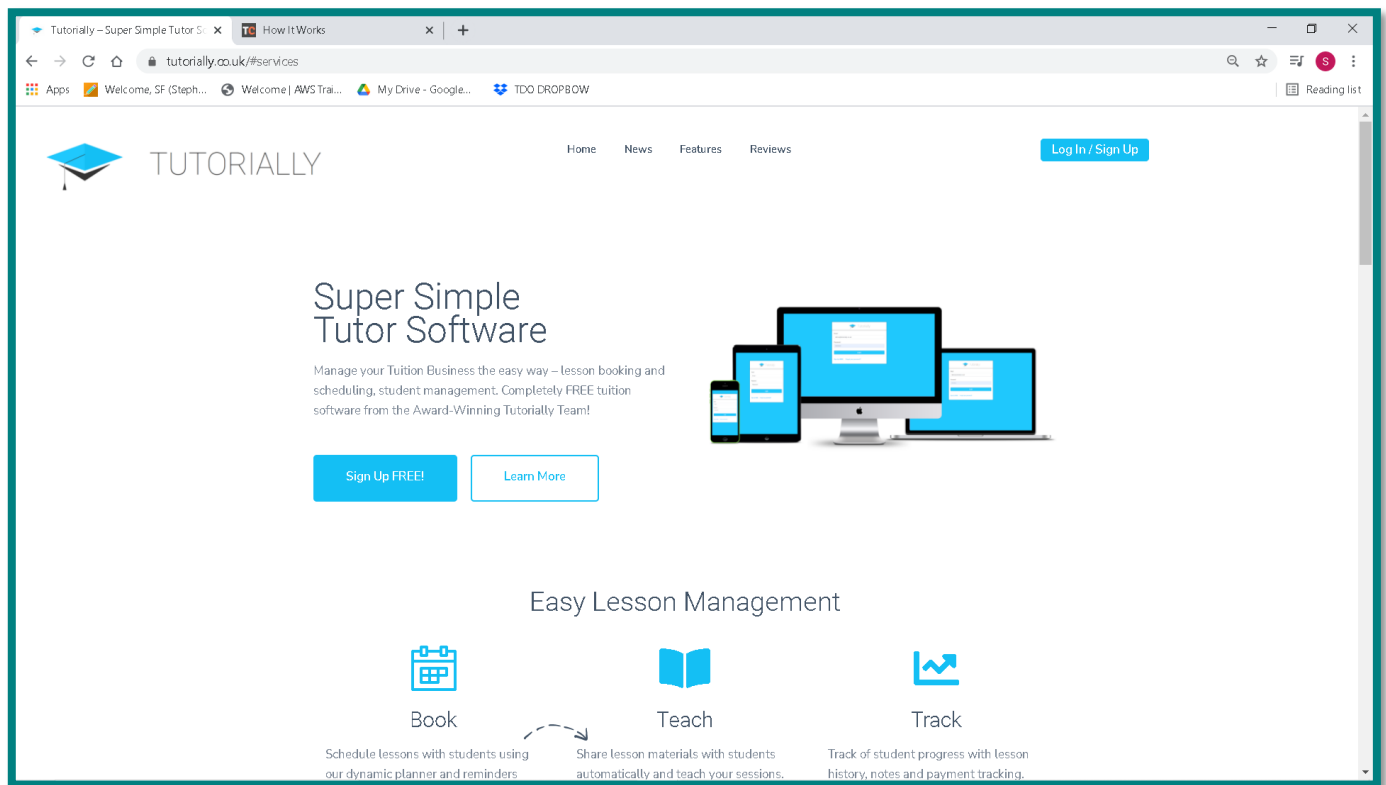


Figure 33 - Tutorially Website Screenshot 1

Tutorially is a super simple tutor software. The upside to this software is that it is completely free to use with no advertisements and no trial period.

Operational features added by Tutorially:

- Detailed lessons – Who, what, where, when – plus notes and file sharing
- Dynamic calendar – Lessons and availability at a glance.
- Student tracking – Detailed student records and progress tracking
- Payment tracking – See who has paid, and who needs reminding.
- Auto Reminders – Lesson reminders reduce no-shows and increase income.
- Tutor profile – Custom Avatars and tutor personalisation.

In the following screenshots We will show what functionality you have access to if you use this software with a free account:

- Dashboard (What it would look like if you had a free account):

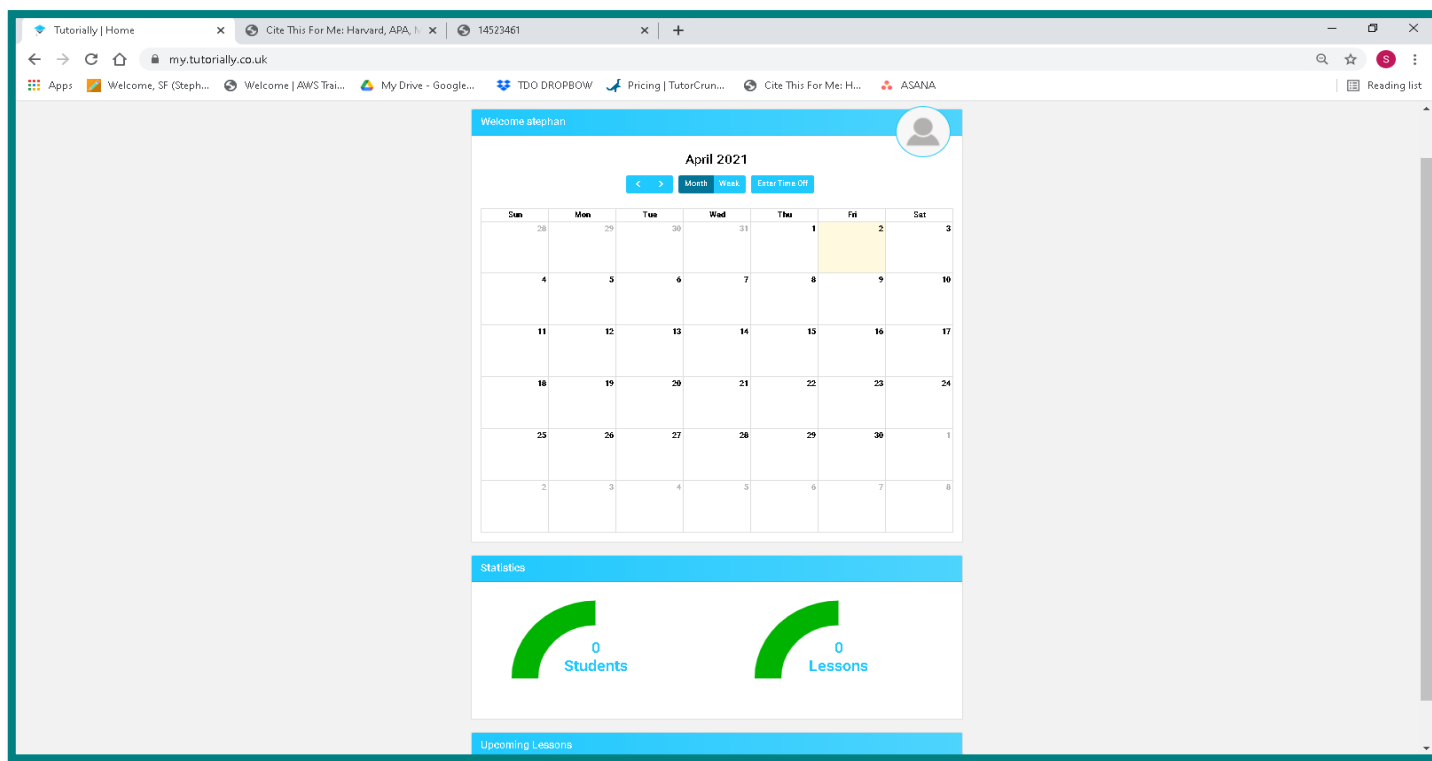


Figure 34 - Tutorially Website Screenshot 2

- Lessons management:

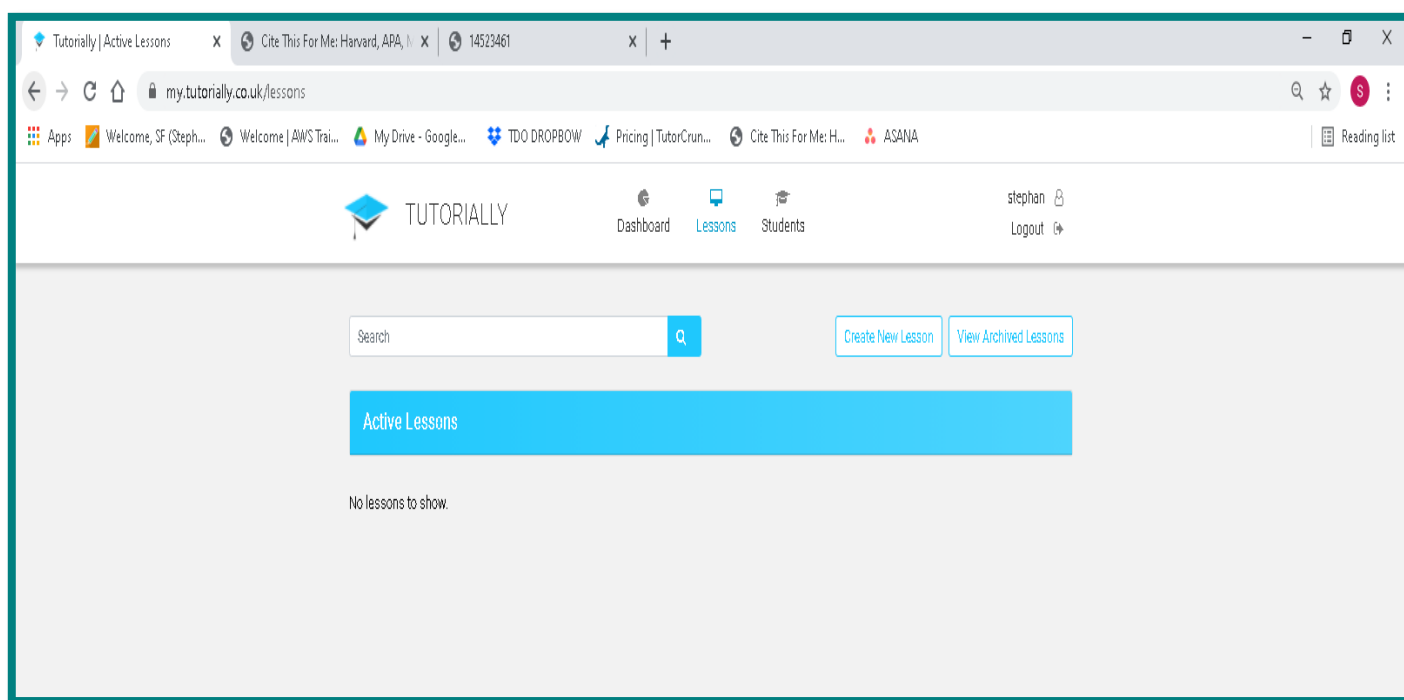


Figure 35 - Tutorially Website Screenshot 3



- Student management:

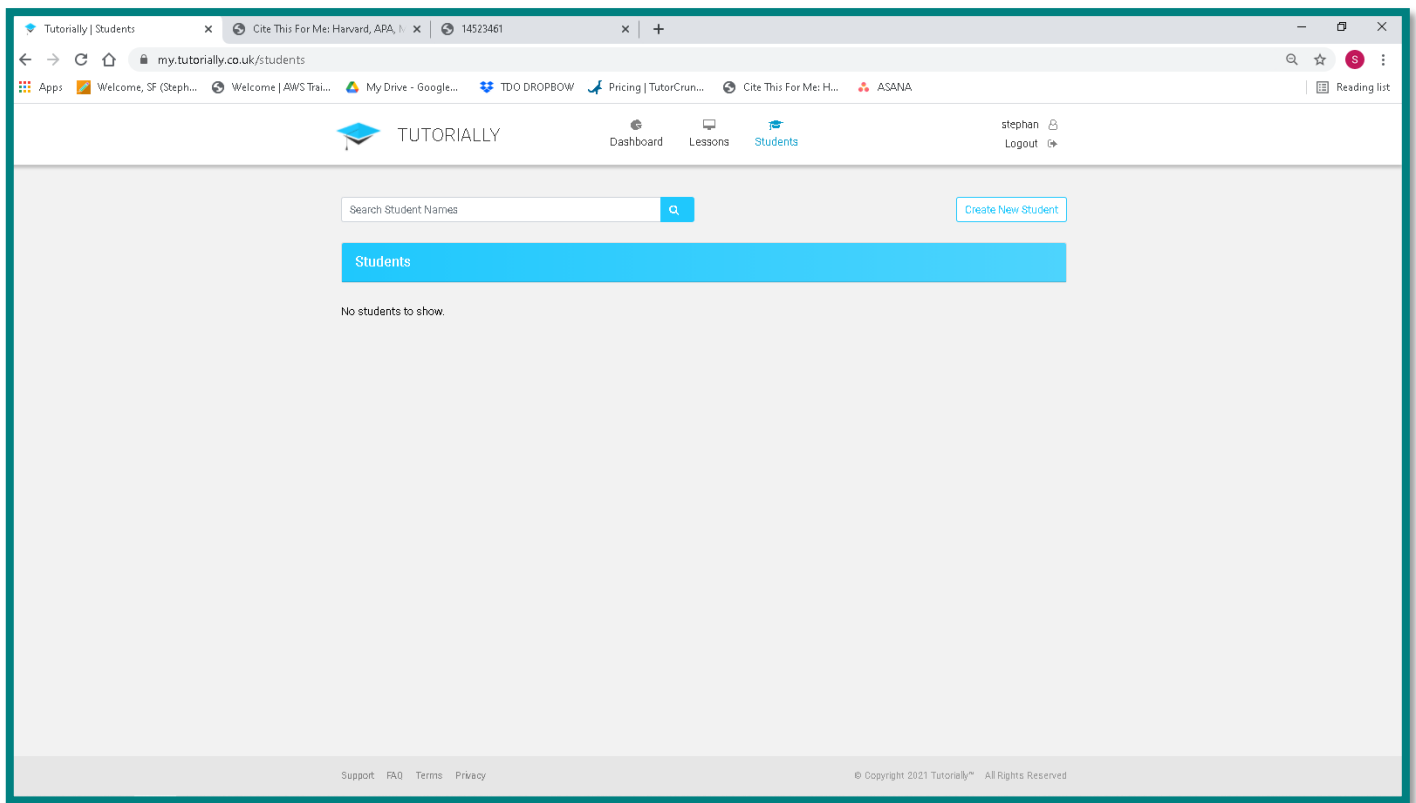


Figure 36 - Tutorially Website Screenshot 4

The reason we did not use Tutorially as a candidate in the feasibility matrix is that it is lacking a lot of functionality and it does not allow for multiple tutor additions and management as well as little to no communication between tutor and student which is crucial.



9.1.6 TeachWorks

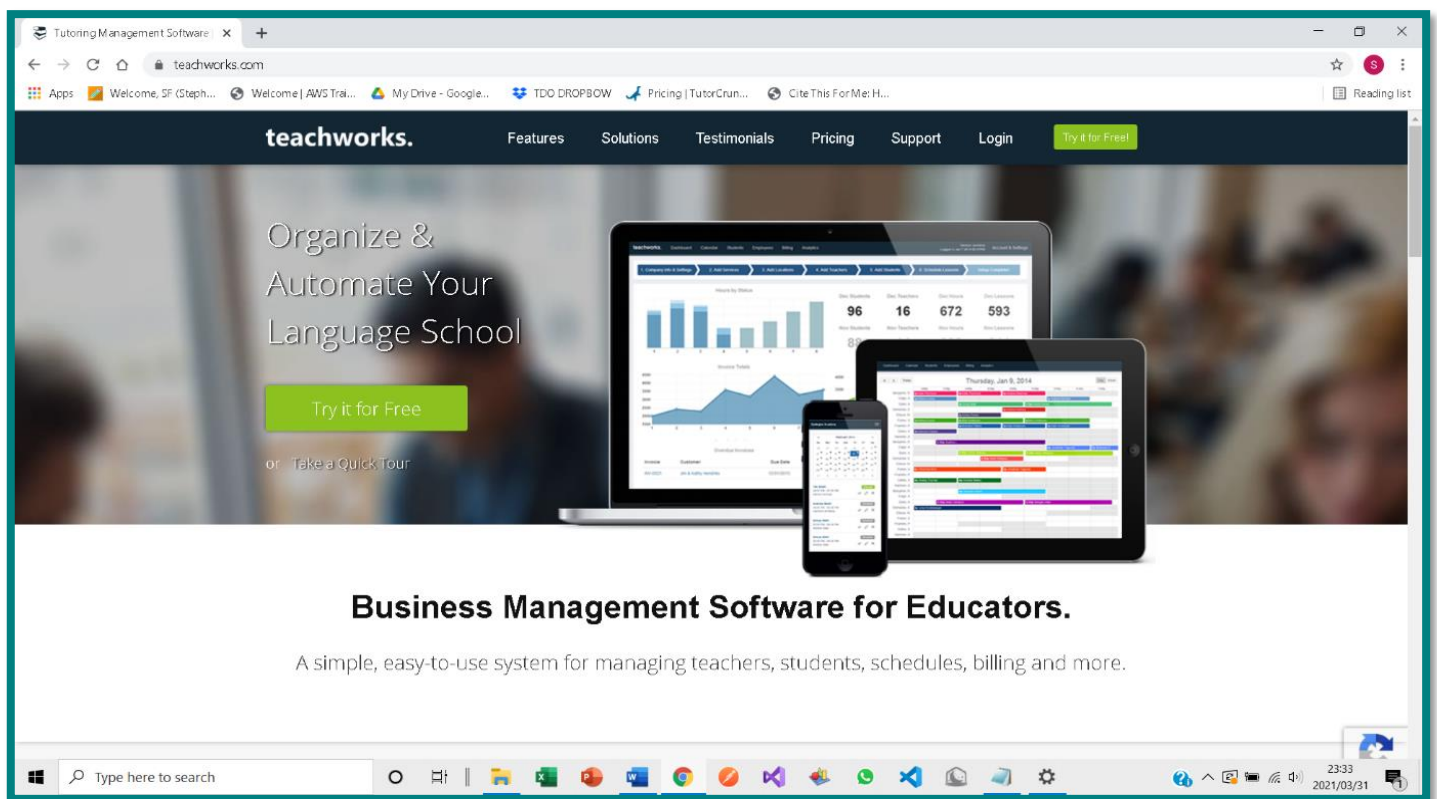


Figure 37 - TeachWorks Website Screenshot 1

Teachworks is a simple, easy-to-use system for managing students, teachers, schedules billing and more.

Operational features:

- Complex scheduling simplified – Coordinating schedules with multiple teachers, students and locations can be a challenge. Teachworks simplifies scheduling and ensures that everyone is on the same page.
 - Multiple calendar views and filters
 - Teacher availability
 - Quick and easy scheduling
 - Spot scheduling conflicts
 - Booking lesson from your website
 - Joining open classes from your website
 - Detailed records
 - Personal calendars



- Enhance & automate communication:
 - Personal, online user accounts
 - Automatic email reminders
 - SMS lesson reminders
- Billing and payroll:
 - Efficient, rule-based billing
 - Automated invoices
 - Credit card payments
- Records and analytics:
 - Maintain detailed records.
 - Record management
 - Custom registration forms
- Customize & white label:
 - Reinforce your brand with each login.
 - Login form to your website
 - Company name on each page
 - Account setting customization.
- Manage multiple locations:
 - Consolidated results in real time
 - Standardized processes
 - Sync to QuickBooks.
- Pricing:
 - Starter - \$15 per month + \$0.25 per student lesson monthly
 - Growth - \$45 per month + \$0.15 per student lesson monthly
 - Premium, - \$175 per month + \$0.05 per student lesson monthly

Evidence of research done on TeachWorks:

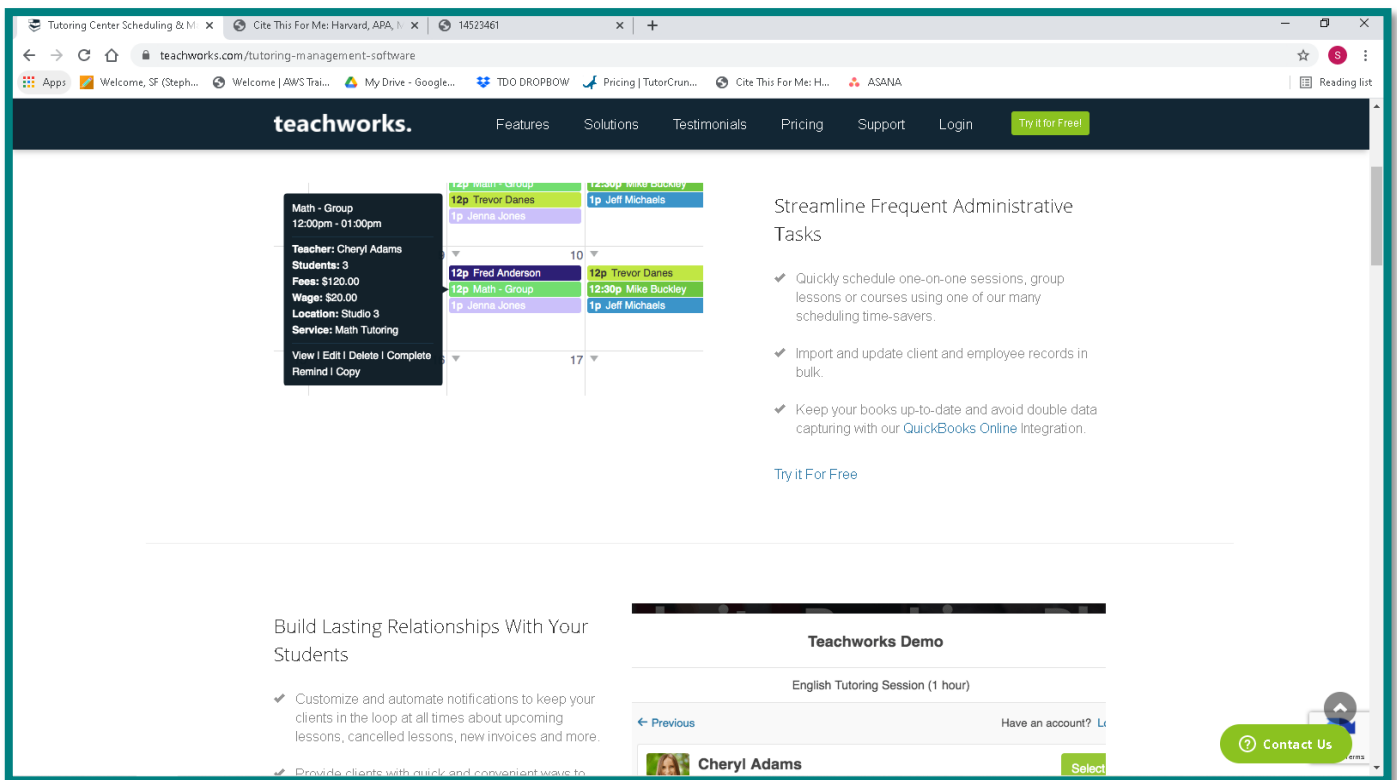


Figure 38 - TeachWorks Website Screenshot 2

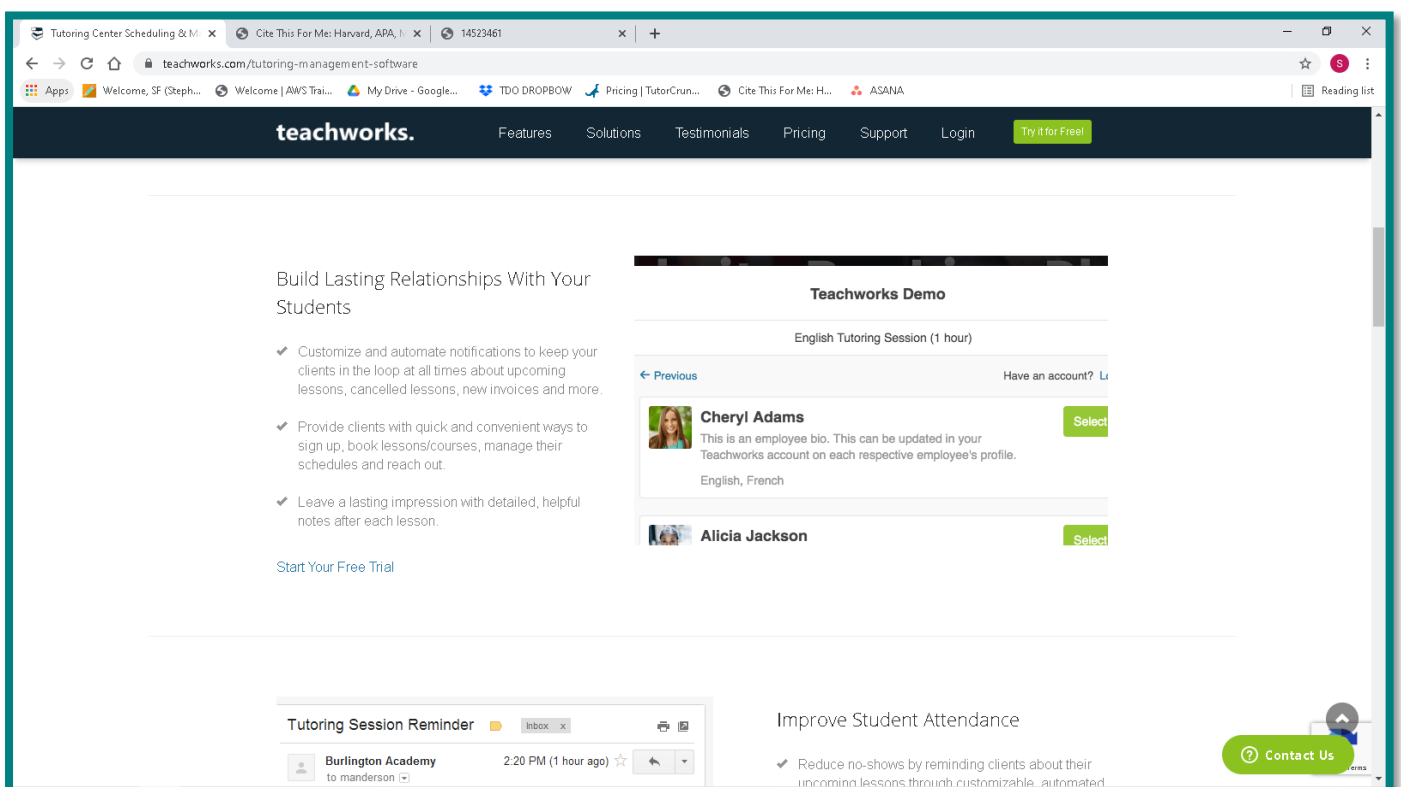


Figure 39 - TeachWorks Website Screenshot 3

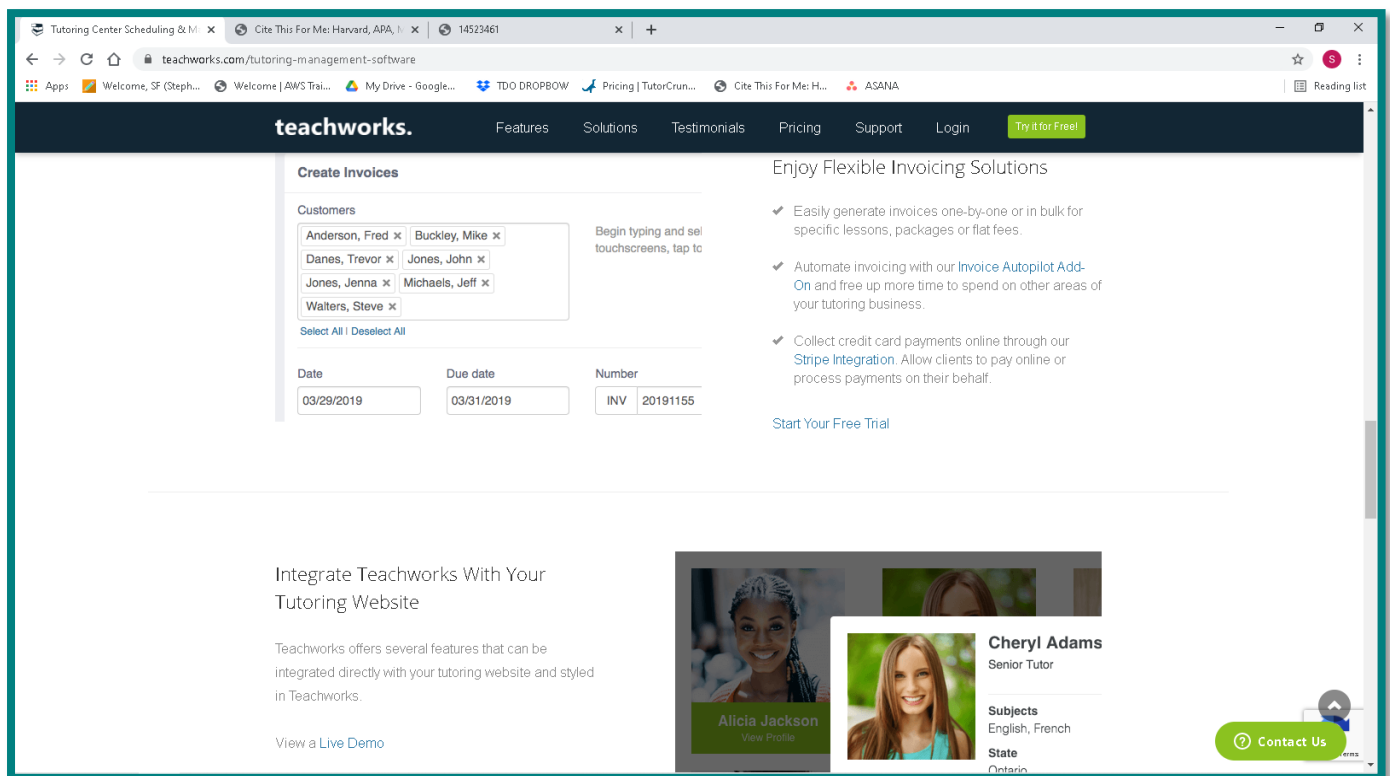


Figure 40 - TeachWorks Website Screenshot 4

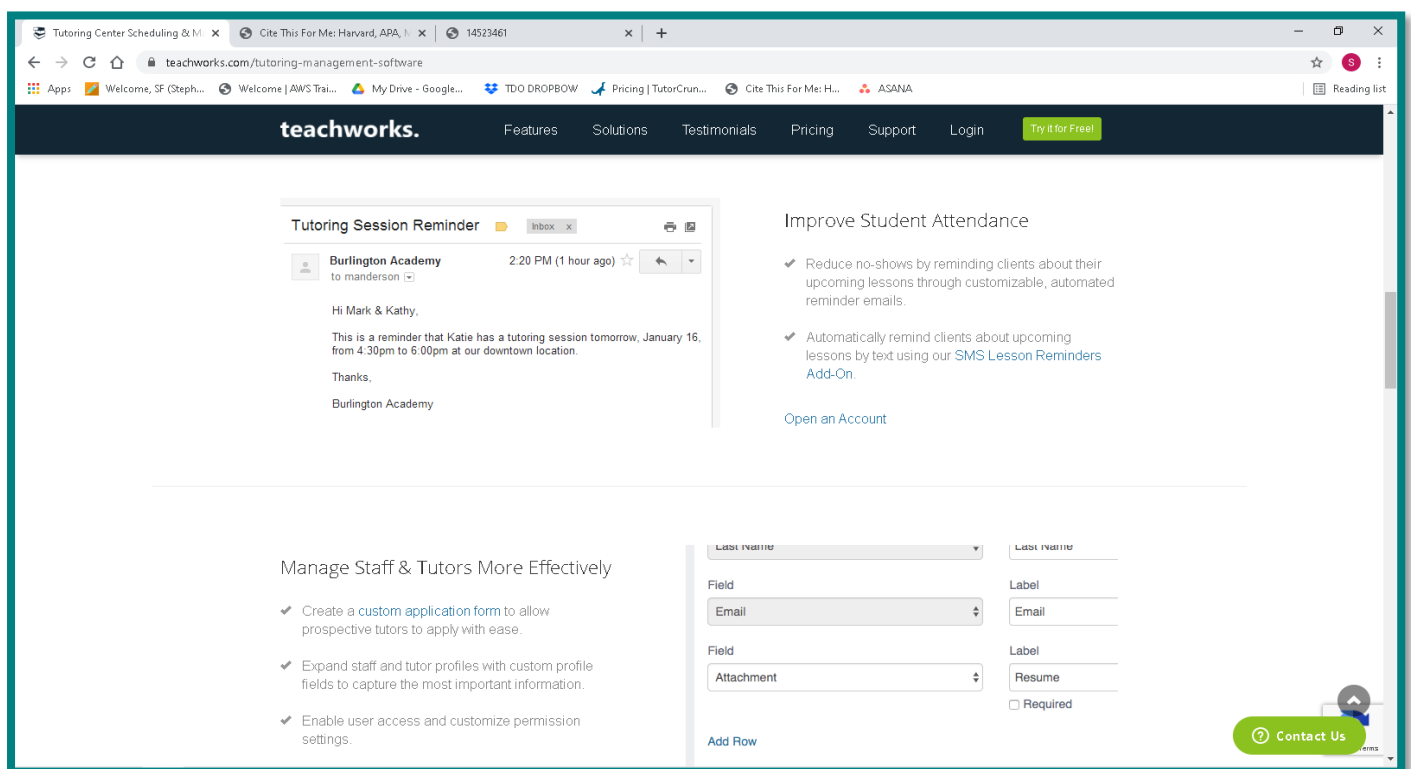


Figure 41 - TeachWorks Website Screenshot 5



9.1.7 Tutor Panel

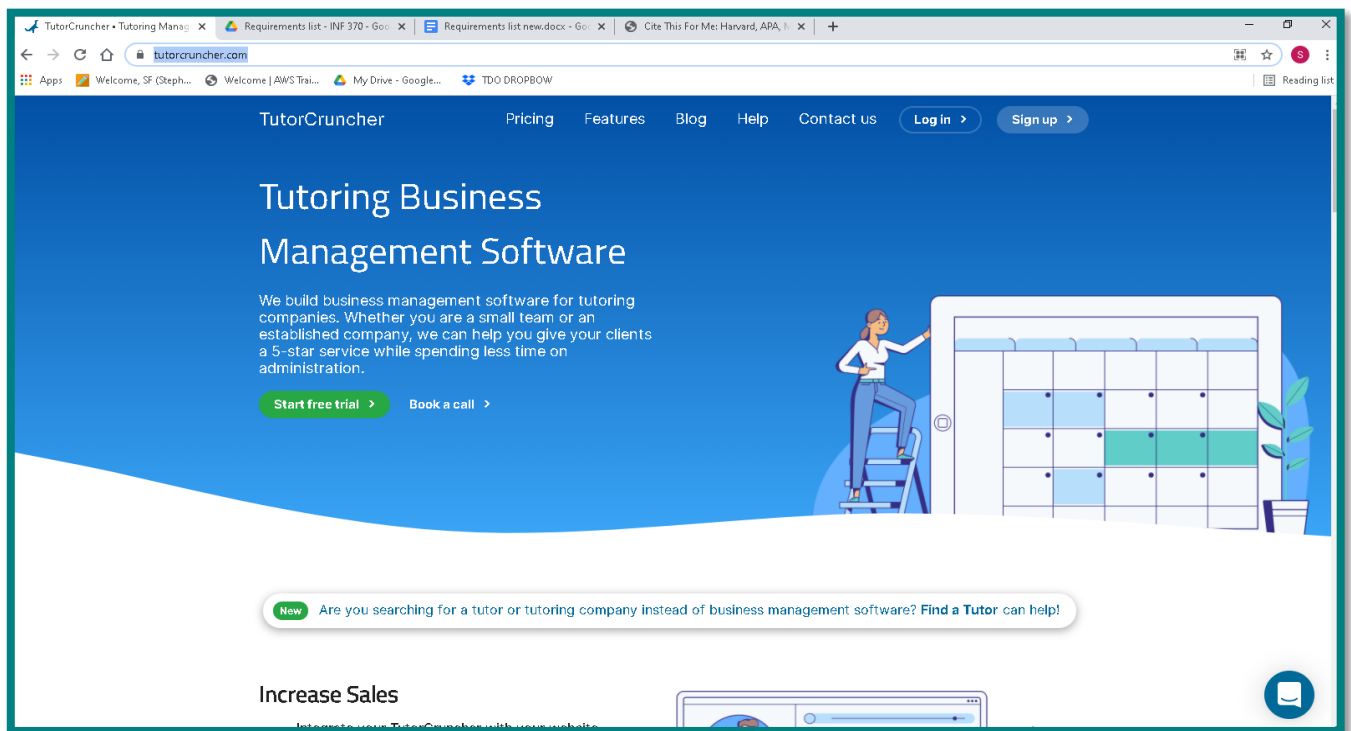


Figure 42 - TutorPanel Website Screenshot 1

TutorCruncher is a tutoring business management software. The TutorCruncher team builds business management software for tutoring companies. Whether it is a small or large business.

Operational features:

- Business analytics

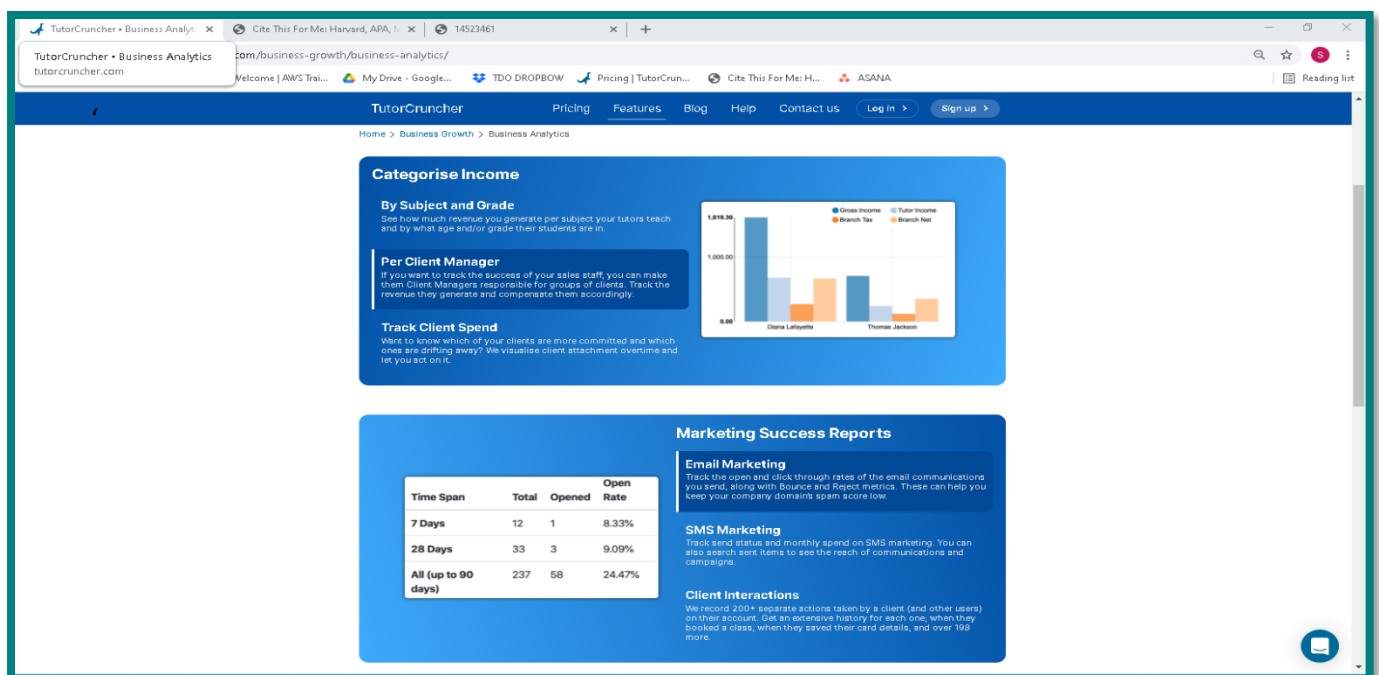


Figure 43 - TutorPanel Website Screenshot 2



- Comprehensive CRM

Profiles

Unique Logins

The option for every Client to access their own personal account where they can save their card details, view reports and [export their calendar](#).

Client Pipeline

Our drag-and-drop client onboarding pipeline is fully amenable to suit your existing sales process. A vital component of any CRM system. [Click to see how it works](#).

Contact History

All emails, SMS, invoices, top-ups and notes related to a client are recorded against their profile for later reference.



Mass Communication

Email

47 Personalised automatic emails triggered by a host of events, time spans and manually. All these emails will be tracked and will be sent from your business email domain.

SMS

17 Customisable SMS alerts, from lesson reminders to interview invitations and invoice notifications. Send texts from your company name.

Broadcasts

[Mass marketing the easy way](#). Send out emails to large groups of users based on your custom categories to inform them about new product offerings, changes in service or other important updates.

Send Credit Request	{{{ company_name }}} Credit Request	✓
Send Credit Request Reminder	{{{ company_name }}} Credit Request Reminder	✓
Send Invoice	{{{ company_name }}} Invoice	✓
Send Invoice Reminder	{{{ company_name }}} Invoice Reminder	✓
Send Paid Invoice	{{{ company_name }}} Invoice Already Paid	✓
Send Payment Order	{{{ company_name }}} Payment Order	✓

Information Sharing

Reviews

Automatically [gather valuable feedback from your clients](#) on how their experience was with your service, use aggregated reviews to evaluate tutors' effectiveness.

Custom Fields

Collect the information on your users that you deem valuable. Want what percentage of your clients are left-handed? Easily done.

Documents

Store and share documents with groups of users or with each individual, you can use this to share learning materials for lessons, contracts for tutors and much more.

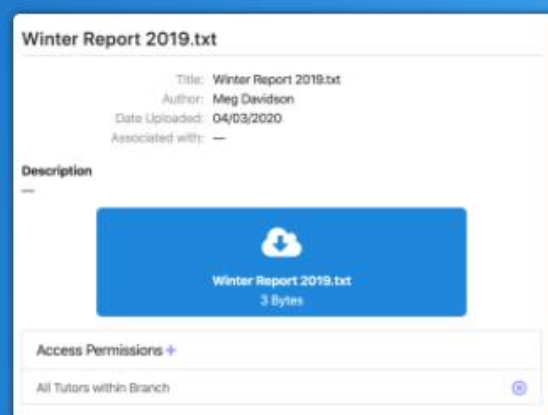


Figure 44 - Mural of TutorPanel Offerings 1



- Intelligent design

Reliable Technology

Safe & Secure

Fully GDPR compliant with gold standard security mechanisms for keeping your clients' data safe including MultiFactor Authentication on logins.

Mobile Optimised

Our entire platform is designed to work fluidly on all commercial smartphones, making for really smooth mobile access.

In Line with Industry Best Practice

TutorCruncher's codebase is managed dynamically to keep it up to date with all the latest advances in SAAS technology.



Constantly Improving



New Features Weekly

New Features Released Weekly

TutorCruncher is always in active development, meaning our developers release exciting new software features every few days. Got a feature request for us? Get in touch [here](#).

Endless Integrations Through Zapier

Zapier connects TutorCruncher to over 2000 other software products. Want to integrate TutorCruncher with your HubSpot account? [Check it out how to setup Zapier](#).

Highly Skilled Software Developers

With our team of serious minded software developers the solution to your software needs is in safe hands.

- Online teaching tools

Broad Capabilities

Video Calls

Voice, Video and screen-share capabilities to connect students and tutors together.

Collaborative Workspace

Have both tutors and student work on a single shared document that allows for file uploads, saving as a PDF, and text and image generation.

Integration Marketplace

Add integrations from an ever-growing number of platforms you can use alongside TutorCruncher.

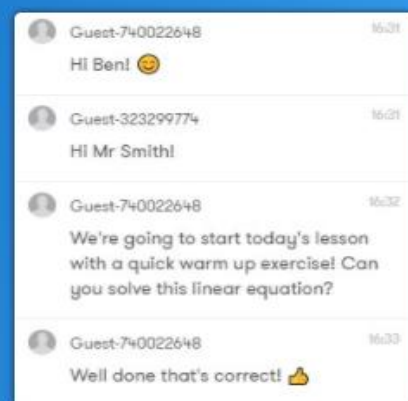


Figure 45 - Mural of TutorPanel Offerings 2



- Scheduling

Intelligent Tutor-Student Matching

By Skills

Selectively advertise job opportunities to tutors based on their skills and experience.

By Location

Only notify nearby tutors of new teaching opportunities in their neighborhoods.

By Custom Category

Have a different metric for how you communicate opportunities? No problem, use labels to notify tutors by price tier, universities attended or even shoe size (if you wanted to).



Plan and Record Lessons

Set Repeating Lessons

No more week-by-week scheduling on spreadsheets. Once you have a time and day that everybody's happy with, set it to repeat long into the future.

Automatic Lesson Reminders

Send email and/or SMS reminders to clients, students and tutors anywhere from 10 minutes before the lesson to two days before. Click to see how to use Lesson Reminders.

Lesson Feedback Reports

After each lesson the tutor is asked to write a unique report to provide feedback, the report format can be fully customised.

GCSE English • 27/04/2020 15:25 1 hour • C • Editable	Planned
Students: 3 • Tutors: 2	
GCSE English • 20/04/2020 15:25 1 hour • C • Editable	Planned
Students: 3 • Tutors: 2	
GCSE English • 13/04/2020 15:25 1 hour • C • Editable	Planned
Students: 3 • Tutors: 2	
GCSE English • 09/04/2020 15:25 1 hour • C • Editable	Complete
Students: 3 • Tutors: 2	

Easy to use Calendar

Extensive Filtering

Colour code and sort your calendar by; Tutor, Student, Client, Client Manager, Course, Location and Status.

External Integrations

With our iCal integration your clients, tutors, students, and admins can access their calendar through Google, Apple or Outlook calendar.

Tutor Availability

We give tutors the option to set their availability so that you always know who you have available when the need arises.

Thu 28/11	Fri 29/11	Sat 30/11	Sun 01/12
11:45 - 13:15 Law tutor required for student	12:00 - 13:30 Microelectronics Semiconductor Industrial	13:00 - 16:00 Economics AS (GCSE) level - Edexcel board • Economics AS Chapter 1	11:45 - 13:15 Law tutor required for student
15:30 - 17:00 Advanced music theory • Advanced			

Figure 46 - Mural of TutorPanel Offerings 2



- Website integration

Ready To Go URLs

Tutor Recruitment

New recruits can sign up independently and submit extensive applications, including their experience, their CVs and the hours they're available to work.

User Login

Direct your clients, tutors and students to log in to their own unique profile, complete with a Password Reset option for those who forget.

Branded To Your Specs

Our custom pages are branded and completely amendable. Want to use your own colour scheme? Write it in with our Custom CSS.

Mockup of a login page. At the top, a blue box contains the text "Your Logo Here!". Below this, a white box contains the text "Welcome to Your Tutoring Company". Underneath, there are two input fields: "Email Address" and "Password". At the bottom right of the white box, there are links for "Reset Password" and "Login Help".

Onsite Plug-ins

Mockup of a client inquiry form. It has four input fields: "Name" (with "Alice Davidson" entered), "Email" (with "Klara Davidson" entered), "Phone number" (with "alice123@email.com" entered), and "Message" (with "I am looking for help with my daughter Klara's GCSEs. She is struggling with her English Lit." entered).

Client Inquiry Form

Use our pre-made form to let new clients contact you easily. On submission, admins will be emailed with the client's information and their profile will be created in the [Client Pipeline](#).

List Tutors Publicly

Show off your top tutors online and let prospective clients pick which ones they'd like to work with.

Open Access Lesson Booking

Your clients can view upcoming available lessons and book themselves in.

Website Building

Experienced Designers

Our designers have created sleek looking websites with great functionality for clients. Get in touch for some examples.

Full Suite Integration

The websites we build come with all the [TutorCruncher plug-ins](#) and URLs for complete connectivity with your management software.

Competitively Priced

We know that website building can cost the earth, so instead we work to a reliable formula and deliver a dependable product that doesn't break the bank.



Figure 47 - Mural of TutorPanel Offerings 3



Pricing:

- 14-day free trial use.
- Pay as you go – R565pm + 1% of revenue
- Start-up – R1350pm + 0.65% of revenue
- Enterprise – Negotiable with the TutorCruncher team (R10 000 000 revenue per year earning prerequisite)

References

- 123tutors. 2021. *Affordable Quality Tutoring | 123tutors | South Africa*. [online] Available at: <<https://www.123tutors.co.za/1-on-1-tutoring/>> [Accessed 2 April 2021].
- Tutorsclass.com. 2021. *Tutorsclass.com: Online Tutoring Platform*. [online] Available at: <<http://www.tutorsclass.com/price/#sthash.Szxbo3eN.dpbs>> [Accessed 2 April 2021].
- Tutorpanel.com. 2021. *Tutor Scheduling & Business Software - TutorPanel*. [online] Available at: <<https://www.tutorpanel.com/>> [Accessed 2 April 2021].
- Wix.com. 2021. [online] Available at: <<https://www.wix.com/>> [Accessed 2 April 2021].
- My.tutorially.co.uk. 2021. *Tutorially*. [online] Available at: <<https://my.tutorially.co.uk/>> [Accessed 2 April 2021].
- My.tutorially.co.uk. 2021. *Tutorially*. [online] Available at: <<https://my.tutorially.co.uk/lessons>> [Accessed 2 April 2021].
- My.tutorially.co.uk. 2021. *Tutorially*. [online] Available at: <<https://my.tutorially.co.uk/lessons>> [Accessed 2 April 2021].
- Teachworks.com. 2021. *Tutoring Center Scheduling & Management Software | Teachworks*. [online] Available at: <<https://teachworks.com/tutoring-management-software>> [Accessed 2 April 2021].
- Tutorcruncher.com. 2021. *TutorCruncher • Tutoring Management Software*. [online] Available at: <<https://tutorcruncher.com/>> [Accessed 2 April 2021].



10. Appendix C: Complexity

Introduction

In the following section the complexity matrix is shown, highlighted in green is the complexity requirements that our system will meet.

Topic	Level		Marks	MAX
1. Special GUI	For online applications: Responsive web design For desktop applications: Form design according to design principles (Schneiderman's golden rule on navigation applies here)	*	3	42
	Appropriate use of grids/tables		3	
	Appropriate use of tabs/links		3	
	Use of graphs in an appropriate business context		4	
	The storage and display of graphical information, like photos with a good business reason		3	
	Working e-mail automatically generated from the database in an appropriate business context		2	
	SMS messages automatically generated from the system in an appropriate business context		2	
	Extensive user-friendly search facility		3	
	At least one use of a tree to display data from the database		3	
	Able to dynamically modify a data tree structure and in doing so adjusting the data in the database		4	
	At least one use of a calendar view of data (not a date/time picker; not a plug-in such as Google calendar)		3	
	Uploading a file into the system with appropriate business reason		3	
	The use of audio/video in an appropriate business context		3	
	At least one use of an administrator configurable timer in an appropriate business context		3	
2. Database access	At least 30 tables used (4 member groups) or 40 tables used (5 member groups)	*	6	15
	Full referential integrity on all tables	*	6	
	At least one use of master-detail table relationships (Schneiderman's golden rule on system status applies here)	*	3	
3. Reports	At least 3 simple list reports in a reporting tool (no control breaks, no graphs, single table)	*	3	15
	At least 2 transactional report with 2 or more control breaks (with heading and calculated values/totals, multiple tables)	*	6	
	At least 1 report with adjustable criteria		3	
	At least 1 management report using a graph		3	
4. Flexibility	All data that can change in future should not be hard coded but maintained in a sub-module of the system (e.g. Lookup tables)		6	12
	Some business rules are not hard coded but maintained in a sub-module of the system.		6	



5. Error handling	All system-generated errors are trapped, and consistent, user-friendly error messages are displayed		6	12
	Appropriate data validation on all input fields		6	
6. Help	At least one menu item or other control that opens up a complete help document (HTML, PDF, Help-file)		3	15
	Extensive context-sensitive help. E.g. calling Help on a specific screen/function will automatically open the specific help for that screen/function.		6	

Topic	Level		MAX	
	Search Facility on Help		3	
	Extensive use of hints		3	
7. Security	Logon screen with user ID and password and fixed user profiles		3	13
	Applying two factor authentication with applicable business reason.		3	
	Encrypted passwords in database	*	1	
	Flexible user profiles (i.e. you can dynamically add user profiles that will enable/disable access to certain parts of the system)		6	
8. Audit Trail	An audit trail of all transactions in the system showing at least date, time, user, transaction type, critical data (such as amount and quantity of transaction)		6	9
	Able to search the audit trail on any of the following: date, user, transaction type		3	
9. Deployment	For a desktop application: Fully functional installation disks that take care of application installation requirements (install and uninstall)		3	15
	For an online application: Deployment of application to a publicly accessible web server		3	
	For a mobile application: Deployment to an App Market place (such as the PlayStore or the AppStore)		6	
	Deployment of the database to a remote database server		3	
10. Backup and Restore	A backup and restore subsystem exists that backup/restore all data (system may exit during restore)		3	3
11. Import/Export Data	Able to open Word or Excel and automatically place data in it based on the parameters provided (with a good business reason)		6	9
	XML or JSON: At least 1 XML or JSON file for Importing or Exporting of data (with good business reason)		3	
12. External INPUT device	Simple Link to an external INPUT device using plug-and-play technology, such as a swipe card reader, bar code reader, etc. or a native component such as a QR reader, a GPS component, etc.		3	18



	Loose Link to an external INPUT device using device specific software. Data or images must seamlessly be stored in the database, but device specific software is visible to the user. (This could include a digital camera, scanner, voice recording device, thump print reader, etc.)		6	
	Tight Link to an external INPUT device using device specific software. Data or images must seamlessly be stored in the database, but device specific software is not visible to the user. (This could include a digital camera, scanner, voice recording device, thump print reader, etc.)		9	
13. External APPLICATION / Services	Integrate an existing web service into your application (with good business reason)		3	9
	A fully functional link to an installed external application system exists and the interface must be shown to work on the external system. Note that this excludes Microsoft Office Applications		6	
Topic	Level		Marks	MAX
14. Multiplatform processing for an appropriate business reason	Appropriate business use of static views on an alternative platform.		3	27
	Appropriate use of dynamic views on an alternative platform (i.e. data is displayed from the system's database)		3	
	Appropriate use of substantial dynamic views on an alternative platform (i.e. both reading and writing data from the system's database)		9	
	Uploading a file through an alternative platform onto the system's database.		3	
	Substantial processing on a third platform (i.e. both reading and writing data from the system's database)		9	
15. Programming Principles	The use of a data layer to facilitate interaction between your database and your business layer		3	12
	The use of an API to facilitate interaction between your business layer and your presentation layer		6	
	Comprehensive use of stored procedures and/or triggers and/or jobs.		3	
16. Innovative addition to the system	Any very advanced innovative addition to the system (e.g. machine learning, AI, block chain, text mining, IOT, etc.)	#	1 - 9	9
TOTAL				157

Conclusion

From the above it can be seen that our group will meet 157 of the complexity marks stated above.



11. Appendix D: Table of Figures

Figure 1 - TDO Logo	7
Figure 2- Map of SA Universities	7
Figure 3 - Calvin Munava, Owner of TDO	9
Figure 4 - Screenshots of Brainstorm Sessions	24
Figure 5 - Mural of Workshop Sessions	26
Figure 6 - Mind map from Brainstorm session.....	29
Figure 7 - Mural of Workshop Session.....	30
Figure 8 - Scan of notes from Workshop Session.....	31
Figure 9- Picture of Current System Ruch Picture	32
Figure 10 - Picture of Proposed Rich Picture	37
Figure 11 - TDO Advert	137
Figure 12 - Mural 1 of TDO Website Screenshots	138
Figure 13 - Mural 2 of TDO Website Screenshot.....	139
Figure 14 - Mural 3 of TDO Website Screenshot.....	140
Figure 15 - Screenshot of TDO Google form	142
Figure 16 - Mural of Welcome Email	143
Figure 17- Screenshot of Welcome email.....	144
Figure 18 - TDO Notification Email	145
Figure 19 - TDO Order confirmation email	146
Figure 20 - TDO Package expiring email	147
Figure 21 - TDO WhatsApp Screenshot 1	148
Figure 22 - TDO WhatsApp Screenshot 2.....	148
Figure 23 - TDO WhatsApp Screenshot 3	149
Figure 24 - TDO Invoice Example.....	150
Figure 25 - 123Tutors Website Screenshot.....	151
Figure 26 - TutorClass Website Screenshot	152
Figure 27 - TutorsPanel Website Screenshot 1	153
Figure 28 - TutorsPanel Website Screenshot 2	153
Figure 29 - Wix Website Screenshot 1.....	154
Figure 30 - Wix Website Screenshot 2.....	155
Figure 31 - - Wix Website Screenshot 3.....	155
Figure 32 - - Wix Website Screenshot 4.....	156
Figure 33 - Tutorially Website Screenshot 1.....	157
Figure 34 - Tutorially Website Screenshot 2	158
Figure 35 - Tutorially Website Screenshot 3	158
Figure 36 - Tutorially Website Screenshot 4.....	159
Figure 37 - TeachWorks Website Screenshot 1	160
Figure 38 - TeachWorks Website Screenshot 2.....	162
Figure 39 - TeachWorks Website Screenshot 3	162
Figure 40 - TeachWorks Website Screenshot 4	163
Figure 41 - TeachWorks Website Screenshot 5	163
Figure 42 - TutorPanel Website Screenshot 1.....	164
Figure 43 - TutorPanel Website Screenshot 2.....	164
Figure 44 - Mural of TutorPanel Offerings 1.....	165



Figure 45 - Mural of TutorPanel Offerings 2	166
Figure 46 - Mural of TutorPanel Offerings 2	167
Figure 47 - Mural of TutorPanel Offerings 3	168

Client Sign-off

Hereby, I Calvin Munava, confirm that I have read Deliverable 1 presented by Installing Inspiration (Team 17, University of Pretoria) and agree to all the information contained in this document, including the project request, preliminary analysis, problem analysis, requirements analysis and feasibility analysis as meeting the necessary requirements. I confirm that all information stated is factual and that I am aware of any and all changes made in this document between drafts.



Calvin Munava

2021/04/11

Date: