
Software Requirements Specification

for

TaskManageable

Version 1.0 approved

Prepared by

Viet-Nguyen Hoang

Pedro da Silva Dergado

Fariba Mahboub Rezaei

Maryam Afshar

Group 4

05/12/2022

Table of Contents

Table of Contents	ii
Revision History	iii
1. Introduction	1
1.1 Purpose	1
1.2 Document Conventions	1
1.3 Intended Audience and Reading Suggestions	1
1.4 Product Scope	1
1.5 References	
2. Overall Description	2
2.1 Product Perspective	2
2.2 Product Functions	2
2.3 User Classes and Characteristics	2
2.4 Operating Environment	2
2.5 Design and Implementation Constraints	3
2.6 User Documentation	3
2.7 Assumptions and Dependencies	3
3. External Interface Requirements	3
3.1 User Interfaces	
3.1.1 Home page with sign-in and sign-up button	4
3.1.2 Sign In Page	5
3.1.3 Sign Up Page	6
3.1.4 TaskManageable Page	7
3.1.5 Assignments Have Been Done	8
3.1.6 About TaskManageable	8
3.1.7 Help Page	9
3.1.8 Chat Online	10
3.1.9 Profile Page	10
3.1.10 Email Reminder Sample	11
3.1.11 Website Layout	12
3.2 Hardware Interfaces	14
3.3 Software Interfaces	14
3.4 Communications Interfaces	15
4. System Features	18
5. Other Nonfunctional Requirements	24
Appendix C: Stakeholder Register	25
Appendix D: interview questions	26
Appendix E: To Be Determined List	29

Revision History

Name	Date	Reason For Changes	Version

1. Introduction

1.1 Purpose

The purpose of this project is to help the students with assignment due dates so that they can check and submit assignments on time.

This project will help them to see all their assignments, tasks, quizzes, and tests/exams due dates in a single place which will help save time.

1.2 Document Conventions

Bold words are for Topics, Subtopics

Italic words are for content

Numbers for Subtopics

The diagrams for database

The diagram of functionality

1.3 Intended Audience and Reading Suggestions

The main intended audience is Students. But it will help other people too who have deadlines for their work.

An audience like developers, and managers who have due dates for their assignments and their work can use this website to help them see all their work in a single place.

1.4 Product Scope

Having all task delivery dates in one place with reminders can greatly improve the reliability not only to students, but office workers as well.

Being a useful tool, it can become very famous among students, and they will not be losing any unnecessary marks for late submission of their work.

Audience can add a new task and its due date, can modify if anything goes wrong, can tick as done and can delete a task if no longer needed.

1.5 References

Brightspace Pulse App -

https://play.google.com/store/apps/details?id=com.d2l.brightspace.student.android&hl=en_CA&gl=US

E.centennial website integration was also an inspiration

2. Overall Description

2.1 Product Perspective

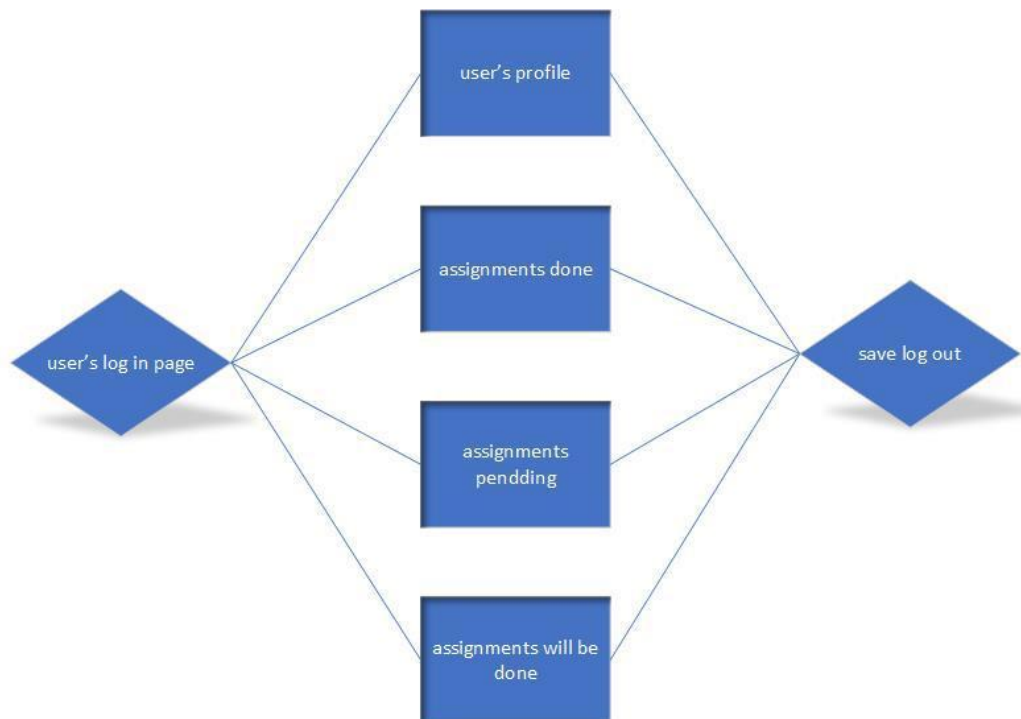
The TimeManageable WebApp is going to be designed to solve the student's problem of noting down all assignments with due dates and crossing them out easily to save their time.

The program has 2 major sections:

1. Tasks recording section.
2. Calendar-like section for deadlines and reminders.

This website is going to be a new product, not a replacement or completing an existing website

The diagram of how the website will work:



2.2 Product Functions

Create an account.

Log in to the account.

Make changes to the profile page such as changing the username, profile picture, and course information.

Insert tasks to be completed and their deadlines.

In the "assignments pending page" can add more assignments and when checks are done they will be added to the assignments done section, add notes, delete and add other assignments with dates.

The user can mark assignments as finished, add the grades achieved, save and delete, and add extra notes.

The user can connect to an integrated API of a third-party softwares to automatically appoint their reminder.

Livechat available to address questions regarding the software.

Emails will be sent to the user reminding them about incoming deadlines.

Users can edit and save all changes.

2.3 User Classes and Characteristics

Mostly the users will be college, university, or school students in educational level. For example, an educational instructor can organize their school project which will be assigned to the students. The technical experience of the users will not matter since the website will be straightforward and user-friendly.

Also, anyone who aims to organize their work can use the product.

The most important and frequent users of the website will be educators.

2.4 Operating Environment

It will be running on a server and the website will be hosted and will be accessible through any browser and any different sized screens such as mobile phones, tablets, desktops, etc. The program will be written in HTML, CSS, backend MySQL.

2.5 Design and Implementation Constraints

2.6 User Documentation

2.7 Assumptions and Dependencies

Users will have to manually enter the assignments and their deadlines (which will be solved in future versions).

3. External Interface Requirements

3.1 User Interfaces

3.1.1. Home page with sign-in and sign-up button



HOME ABOUT HELP SIGN IN SIGN UP

TASK MANAGEABLE

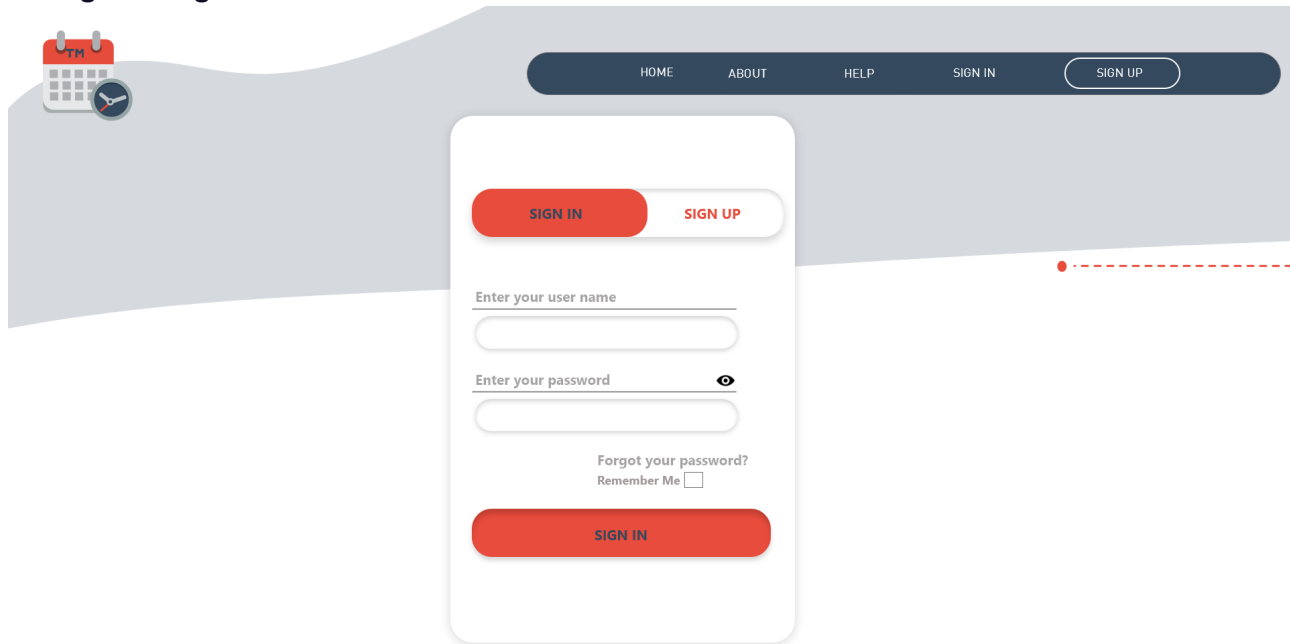
SIGN IN

SIGN UP



The home page, which is the start page, has enough white spaces with two SIGN IN and SIGN UP buttons in dark orange color that are easily visible and attract the user's attention and also the user does not need to look for them. They are easily accessible, both in the navigation bar and in the main body of the page.

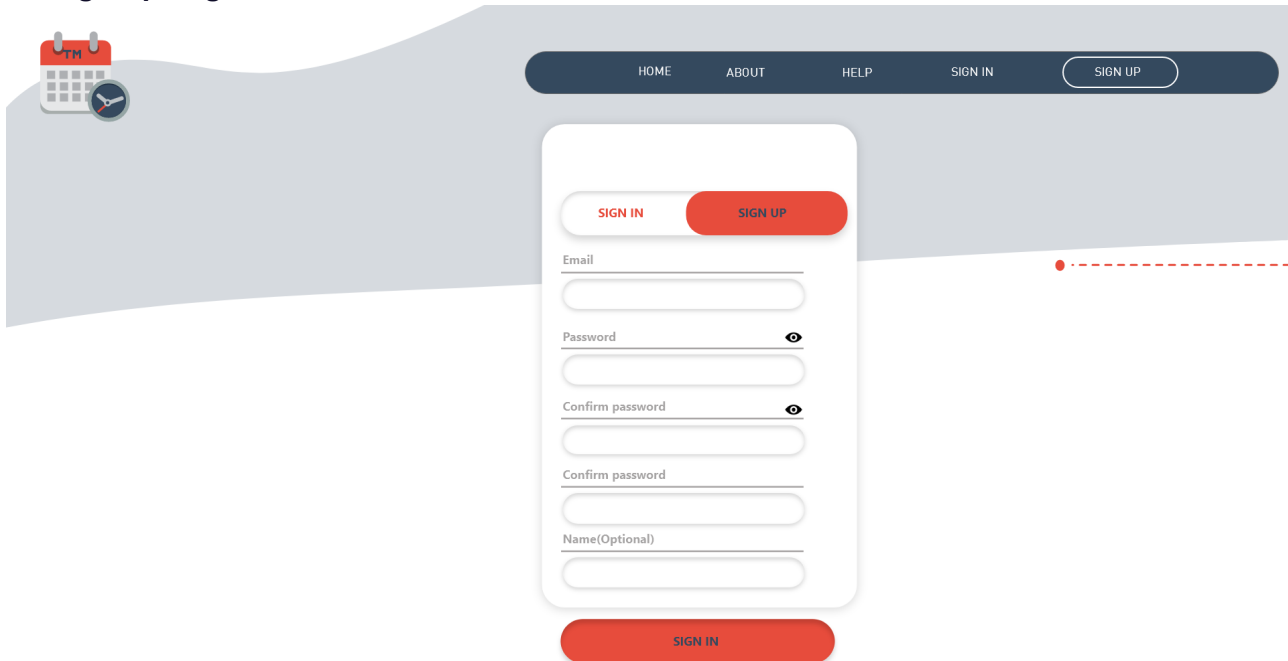
3.1.2. Sign In Page



In the Sign in page, users who already have an account can log in. This page is easy to use and there are again enough white spaces. In this page there are:

- a) Username input
- b) Password input
- c) Sign in button
- d) Forgot password button that helps to change the password
- e) Remember me button so that the user wouldn't have to enter the username and password all the time they want to log in.

3.1.3. Sign Up Page



This page is for new users to open an account.

In this page there are:

- a) *Email input*
- b) *Password input*
- c) *Confirm Email input*
- d) *Confirm password input*
- e) *Name which is optional*
- f) *Username may be added as well. The username will appear on the profile page*

3.1.4. TaskManageable Page

The screenshot displays the TaskManageable page interface. At the top right, a dark navigation bar contains links for HOME, ABOUT, and HELP, along with a user profile section showing 'WELCOME Anna' and a notification bell with a '1' indicator. On the left side, there is a calendar icon with 'TM' and a clock icon. Below these icons is a form titled 'Add your assignment here' with three input fields: 'Add your assignment here', 'Related course', and 'Due date:'. A red 'ADD TO LIST' button is positioned at the bottom of the form. To the right of the form is a 'To do list:' box containing a table with the following structure:

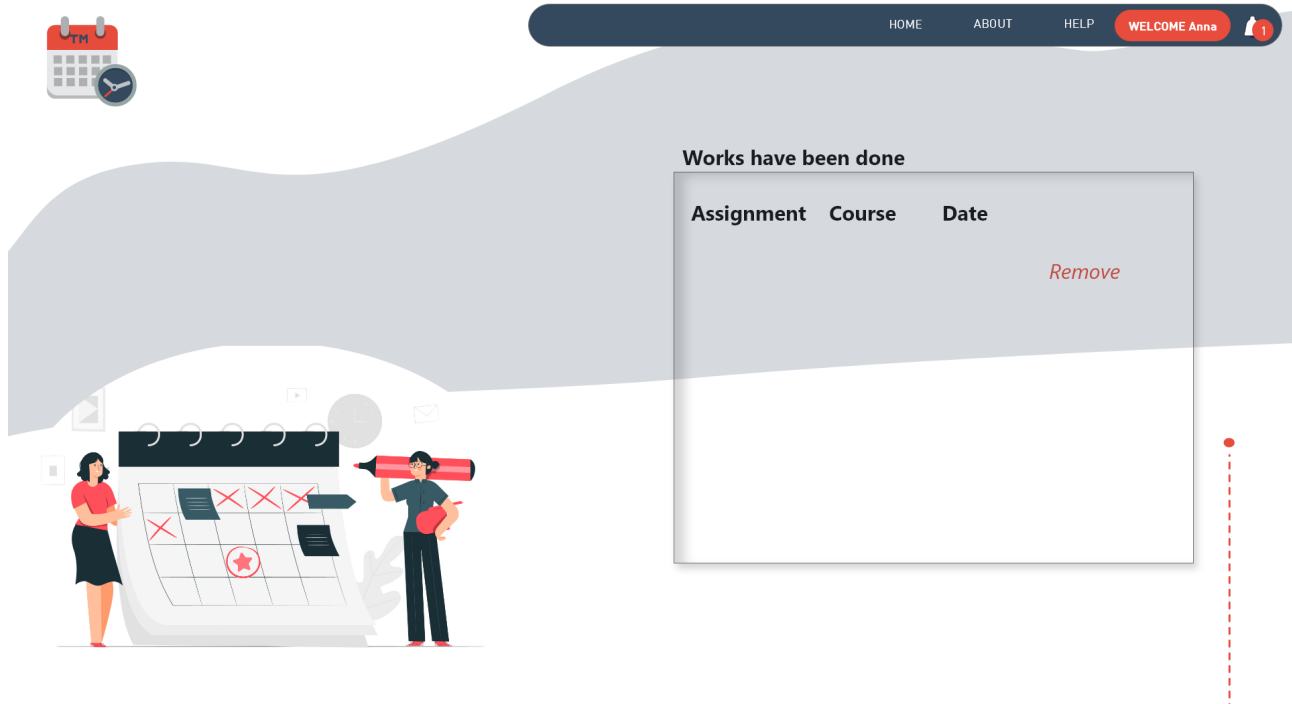
Assignment	Course	Date	Mark as done
			Remove

A vertical dashed red line is located to the right of the 'To do list' box.

This is the main functional page of TaskManageable. In this page there are:

- Adding assignment input: The user can add the name of the assignment
- Related course input: The user can add the name of the course of the assignment
- Due date input: which is the due date of the assignment
- Add to list button: When the user fills the inputs and clicks this button, the information will be added to the to do list box.
- To do list box: The box in which all assignments added will be listed. Assignments will be under assignment headline, courses under course headline and dates under the date headline.
- Mark as done: When the user marks the assignment as done, it will be added to the list in the other page
- Remove button: When the user clicks on the remove button, the assignment will be crossed out.

3.1.5. Assignments Have Been Done



In this page the assignments which are marked as done in the previous page will be added here based on the assignment name, course and date it is done. Also, the user can remove them from the list.

3.1.6. About TaskManageable



About TASK MANAGEABLE

Task Manageable is your handy app for scheduling your assignments efficiently, eliminating the sadness of forgetting the due dates.....



In this page, a description about the purpose of the website will be added. And what functions the page has as well as the advantages of using this website.

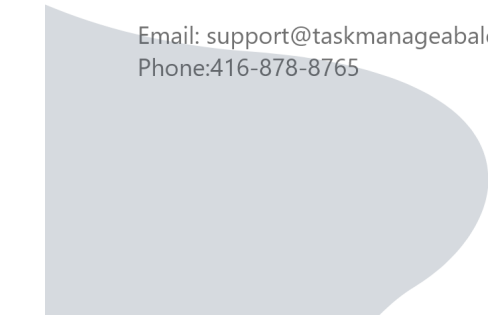
3.1.7. Help Page



You need help?

If you need help about using the app or you have any questions our support team is there to help you. You can contact use by email, phone or chat online.

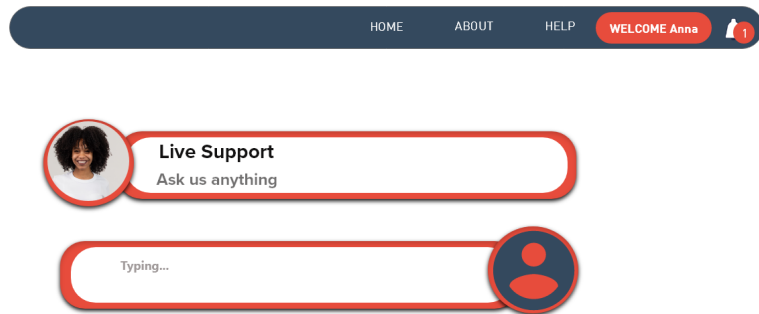
Email: support@taskmanageabale.ca
Phone:416-878-8765



In the help page, user can find information Like:

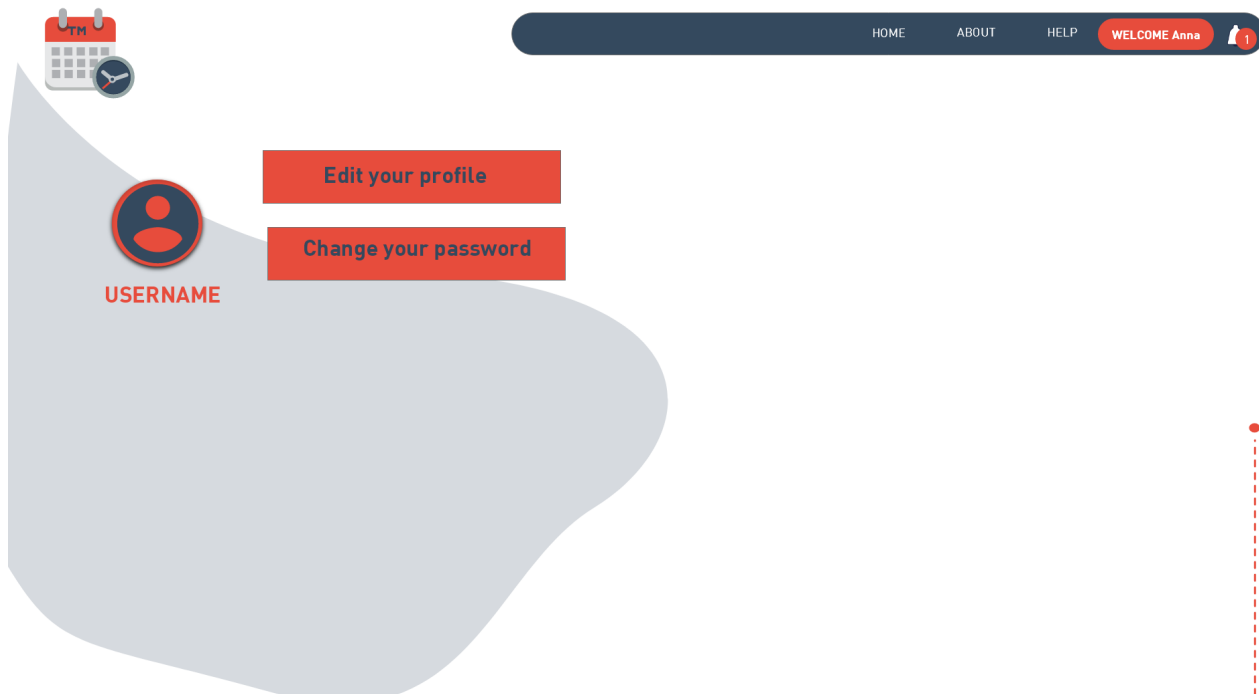
- a) Email address
- b) Phone number
- c) Live chat: when he chooses to chat live, the user will be directed to the chat page.
When there is a website support, the user can trust the page and use it frequently.

3.1.8. Chat Online



In this page the users can chat online and ask questions.

3.1.9. Profile Page



This page is used to customizing the account such as profile picture, username, password.

3.1.10. Email Reminder Sample



Dear Anna,

This is to remind that you have task called "Database Assignment" for course COMP 123 due to September 13.

Task Manageable team

Email: support@taskmanageabale.ca
Phone:416-878-8765

This is a template of the reminder email sent to the user.

3.1.11. Website Layout

a) The color palette:

The main colors which will be used in this website are:



b) The fonts:

Aa

DIN

H1	The spectacle before us was indeed sublime.
H2	The spectacle before us was indeed sublime.
H3	The spectacle before us was indeed sublime.
H4	The spectacle before us was indeed sublime.
H5	The spectacle before us was indeed sublime.

Paragraph fonts: Segoe UI, sans serif, 36, regular

HEADING WITH SUBHEADER

The spectacle before us was indeed sublime.

Subheader

PARAGRAPH

Apparently we had reached a great height in the atmosphere, for the sky was a dead black, and the stars had ceased to twinkle. By the same illusion which lifts the horizon of the sea to the level of the spectator on a hillside, the sable cloud beneath was dished out, and the car seemed to float in the middle of an immense dark sphere, whose upper half was strewn with silver. Looking down into the dark gulf below, I could see a ruddy light streaming through a rift in the clouds.

The font on the logo:



The font on the buttons:



c) Logo: The logo which is edited in illustrator has the initial letters of "TaskManageable on it..



d) The buttons: Most of the buttons will be in orange color with the effect of hover.



The primary button color and size in most of the pages



The same button with hover effect



The main button in sign in and sign up pages
The size is bigger because we want to make the size consistent with the form of the page



The same button with hover effect

e) Consistency: In order to have consistency between the pages we are going to have the wavy background image in all pages just from different angles and in the home page it will be darker but in other pages it will be in lighter color.

The HOME Page:



Other pages(example):



Also, there will be an orange dashed line in all pages:



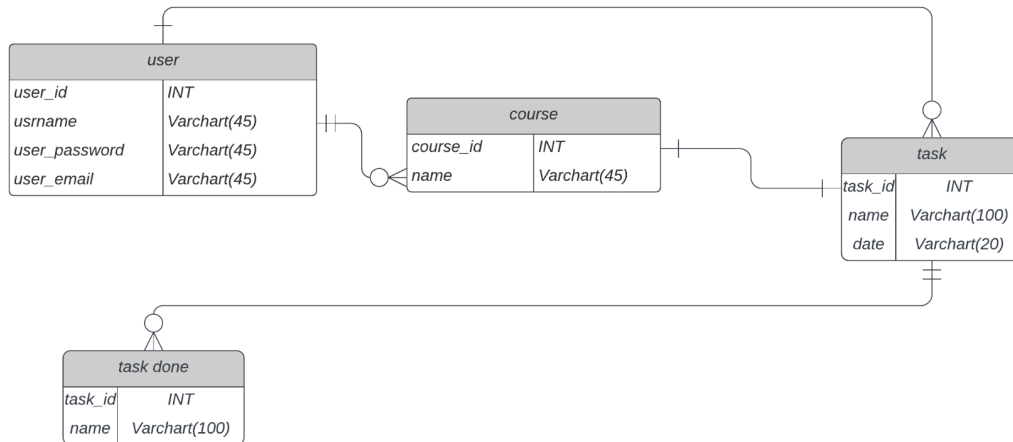
3.2 Hardware Interfaces

None

3.3 Software Interfaces

a) *Front-end programming:* To code the front end of this website we will use HTML5, CSS3, JavaScript and jQuery libraries.

b) *Back-end programming:* For backend, we will use PHP and MySQL. On our website, there are a lot of forms and inputs to be filled by the users, for example, log-in data, sign-up data, profile information, and reminder data. All the data entered by the user should be validated and stored. Since we have not studied how to connect PHP to MySQL, it is not completely clear to us. However, for each user we may have the following ERD for the database relationship:



c) **Fonts:** We will use google fonts and include the links from the google font website <https://fonts.google.com/>

d) **Code Editor:** We will use Visual Studio Code.

e) **Wireframing and prototyping:** We used XD for prototyping and wireframing.

For Images and logos: We will use illustrator and photoshop. The wavy background is also going to be done in the illustrator. The SVG image used is from <https://www.pixeltrue.com/free-illustrations>, which is designed so that we can use CSS animation to animate some parts of the illustration.

3.4 Communications Interfaces

a) **Email system:** When a user signs up, he receives a notification for email verification.

b) **Notifications on site:** The users will see notifications of deadlines on the site when they log in.

c) **Notifications through Email:** The system will send notifications of deadlines to the users' emails.

d) **Contacting System:** On cell phones by clicking on the phone number, the user can call us. Also, the user will be able to send an email by clicking on the email address.

e) **Online support:** There will be a team member at certain hours to support the users through a live chat room.

Use cases			
Use Case name	List of related Requirements ID	Actor(s)	Brief Description
Create user account	FR02	Customer	To create an user account, the user first clicks on the sign-up button in the navbar or in the landing page under the Heading, then the user will be directed to the sign-up form. The user has the options to fill the 'user name', 'email', 'password', and 'confirm password' fields. After submitting all information the user can click on the sign-up button to create the account.
Sign in	FR02	Customer	The user enters username and password, then click sign in to sign in. If sucessfully signed in, the user will be redirected to their account profile.
Add tasks	FR01	Customer	On the task page, the user has the option to add new tasks, after clicking on it there is a subject, types of the task, and due date. Then he can click on the save button. The task will be added.
Add task automatically by connecting to third-party software	FR11	Customer	Users can go to their profile page, underneath the "Change Password" option will be a list of third-party software that they can connect with.

			These software includes Outlook, Gmail, and D2L-based school websites. Users can connect to one or many, duplicate reminders will be automatically sorted. This will enable the users to add reminders manual-free at ease.
Access the Alerts on the task page	FR03, FR10	Customer	Users should click on the bell symbol in the navbar, then they are able to see the alerts for the future due dates there.
Enable/Disable Alert via Alert center	FR10	Customer	
Adding Frequently asked questions	FR06	Investor	The investor will have a slightly different UI than the customer. When accessing the FAQ page, apart from the already initialized question, the investor will be able to see an edit button at the top right corner of the page. When clicking that button, the system will allow the investor to change, edit, or delete any question on the FAQ page.
Access the live chat	FR07	Customer	The user will click to access Livechat to address questions regarding the software. The Livechat function will utilize a chatbot to manage and redirect the questions accurately.
Change password	FR09, FR04	Customer	There will be a button

			that allows the user to click, the “Forgot your password?” on the sign-in page. The user should receive an email containing a temporary passcode and enter the code to access it. Once logged in, the “Change password” option on the profile page will allow the user to update the password.
--	--	--	--

4. System Features

Functional Requirements list				
<u>Requirement ID</u>	<u>Requirement title</u>	<u>Short Description</u>	<u>Priority</u>	<u>Requester</u>
FR01	Task management	The system should have functionality for writing the tasks with due dates and the functionality to remove the tasks. The user can also specify how many times and how many days are left till the due date to place an alert	High	Investor
FR02	User accounts	The system should have the sign-in/ sign up features to know how many	High	Investor

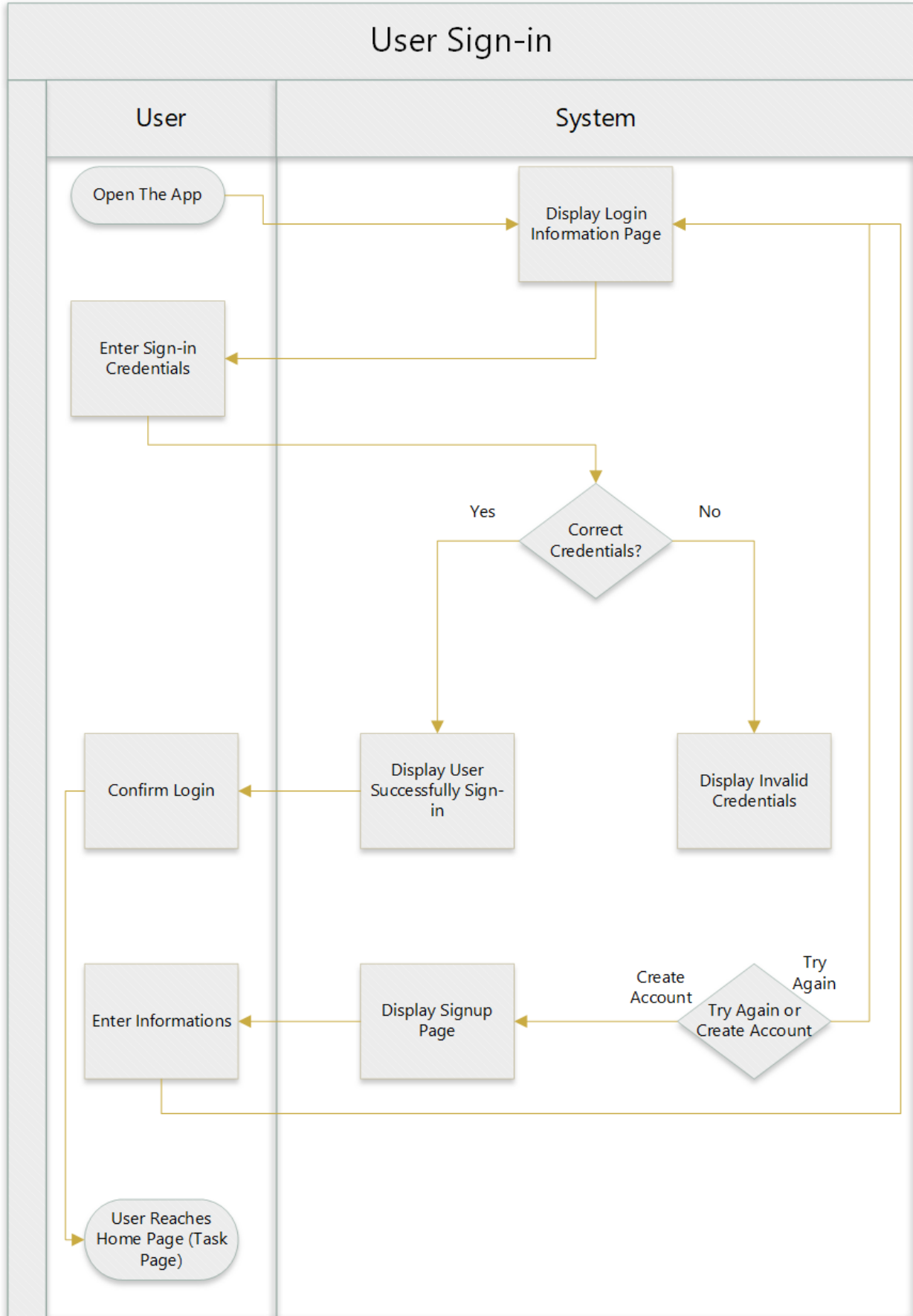
		users the system has and protect their credentials		
FR03	Alerts	The system should give the user notification/alert about any upcoming due dates	High	Customer
FR04	Profiles	The system should allow the user to update their profiles (Name, Profile Picture, Password).	Medium	Programmer
FR05	E-mails	The system should send email notifications to the user about any upcoming due dates	High	Customer
FR06	Frequently asked questions FAQ	The system should have the capability to allow the investor to set up and manage frequently asked questions.	Medium	Investor
FR07	Support function	The system should have help functionality through email and online-chat	Medium	Customer
FR08	Remember user credentials	The system should have a choice at the sign-in page to ask whether or not the user wants to save their credential so next time they don't need to enter again	Medium	Customer
FR09	Forgot password or	The system should	High	Customer

	username	have a button at the sign-in page for the user to reset their password or remind them of their username by entering their registered email		
FR10	Alerts center	The system should have an alerts center so that the user can easily manage and review their alerts. The user can also enable/disable the alert in the alerts center	Medium	Programmer
FR11	Integrated with the third-party application	The system should have an option for the user to connect with third-party software to automatically set their reminder	High	Investor

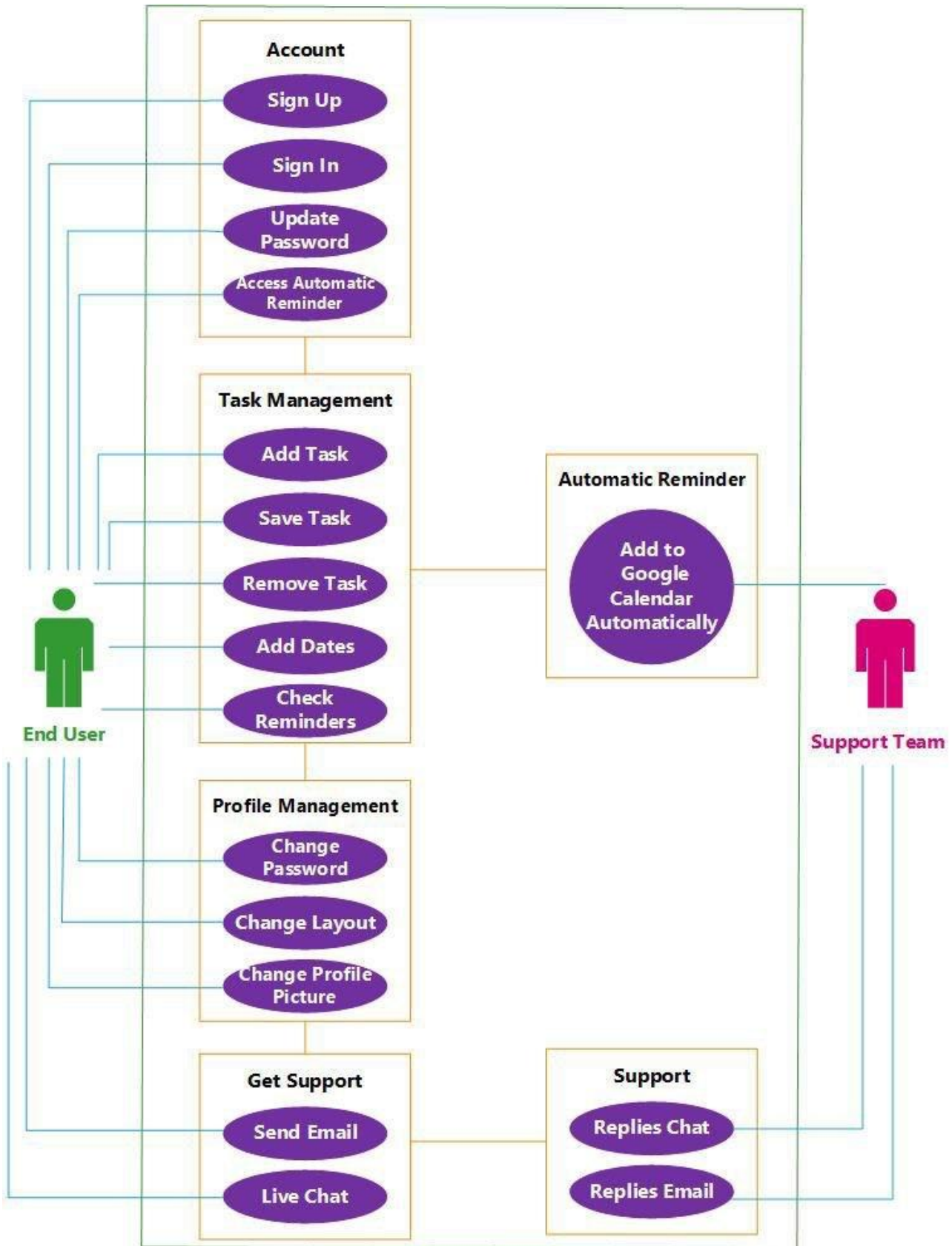
USE CASE		
Use case name	Sign in to the system	
Primary actor	The end-user (student)	
Goal in context	Sign in the system to work with the application	
Preconditions	Having valid account credentials (username and password)	
Trigger	Sign in button	
Scenario details	Actor	System
	<ol style="list-style-type: none"> 1. Open the app 2. clicks on sign in button 3. Enter the valid username and password 4. Receives success message 5. Logs in the system 	<ol style="list-style-type: none"> 1. Server receives input 2. Verifies the sign-in credentials 3. If valid, sends success verifications 4. Logs the user in the system 5. If not valid, sends “not valid information” message
Exception	If the user types in the wrong password or username an error message will return: “the username or password provided does not exist, please try again or request email verification for password change”	

Priority	High
-----------------	------

When available	All the time
Frequency of use	Frequent
Channel to actor	Web Browsers such as Chrome, Edge, Safari,...



Use Case Diagram:



5. Other Nonfunctional Requirements

<u>Nonfunctional Requirements list</u>				
<u>Requirement ID</u>	<u>Requirement title</u>	<u>Short Description</u>	<u>Priority</u>	<u>Requester</u>
NFR01	Response time	The data system shall show no visible deterioration in response time as the number of people increases. Response times seen by end-users for retrieving the list of to-do tasks or completed tasks should take a few seconds or less.	High	IT Manager
NFR02	Responsiveness (screen-size compatibility)	It should work well in different screen sizes such as desktop, tablet and mobile	High	IT Manager
NFR03	Browser compatibility	The system should work well in most of frequently used browsers such Chrome, Firefox, Safari	High	IT Manager
NFR04	Confidentiality	The system should be secure enough for the users to store their personals tasks (like hashing the password or sanitizing the data)	High	IT manager
NFR05	Accessibility	The system should pass accessibility testing (like color contrasts, font size, adding 'alt' element to images while	High	Investor

		coding,...)		
NFR06	Auto-maintenance	The system should perform a weekly small maintenance automatically to ensure there are no breaches in security or errors.	High	Investor

Appendix C: Stakeholder Register

Stakeholder Register					
Stakeholder name	Stakeholder Position	External /Internal	Stakeholder Contact Detail	Operational/ Executive	Interest (high, medium, low)
Ahreeba Gamble	Customer	External	ahree.gam@gmail.com	Operational	High
Jakub Mcfarlane	Marketing Officer	Internal	jamcfar@cop.com	Operational	Medium
Ayla Macguire	IT Manager	Internal	ayla.mac@cop.com	Executive	Medium
Anne Melia	Investor	External	annemelia@gmail.com	Operational	Medium
Reagan Dresswell	Programmer	Internal	reagan.dress@cop.com	Operational	Medium

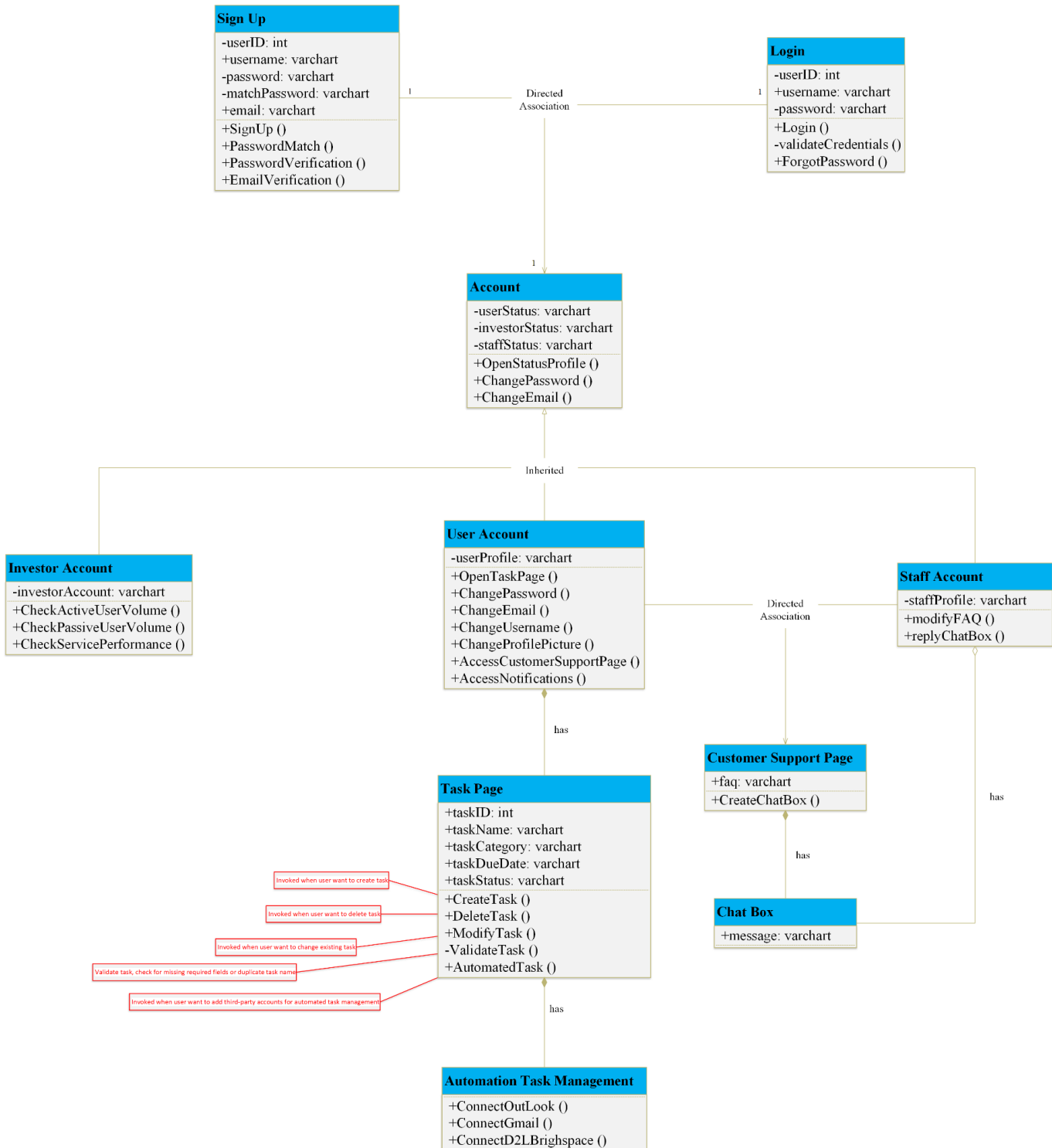
Appendix D: Interview Questions

Interview Questions		
Question	Stakeholder position	Answer
What is the main purpose of the product?	Investor	Acquire many users and promote our company as a solid player in the market.
How much is your budget for this project?	Investor	Initially we are planning an investment of \$30,000.00
What is the general requirement for the WebApp?	Investor	It should be simple, user-friendly, and easy to maintain
What will the product do in general?	Investor	The product should accommodate the customer's need of managing their tasks
When should the product be delivered?	Investor	Ideally before the end of the year, so in around 7 months.
To which countries do you ship products and what languages are supported?	Marketing Officer	North America and English only.
What is the target market that you are aiming for?	Marketing Officer	Our main focus is students at the moment with future plans to target office workers.
How many users are expected to be using the product?	Marketing Officer	We are estimating a maximum of 5000 concurrent users at any time.

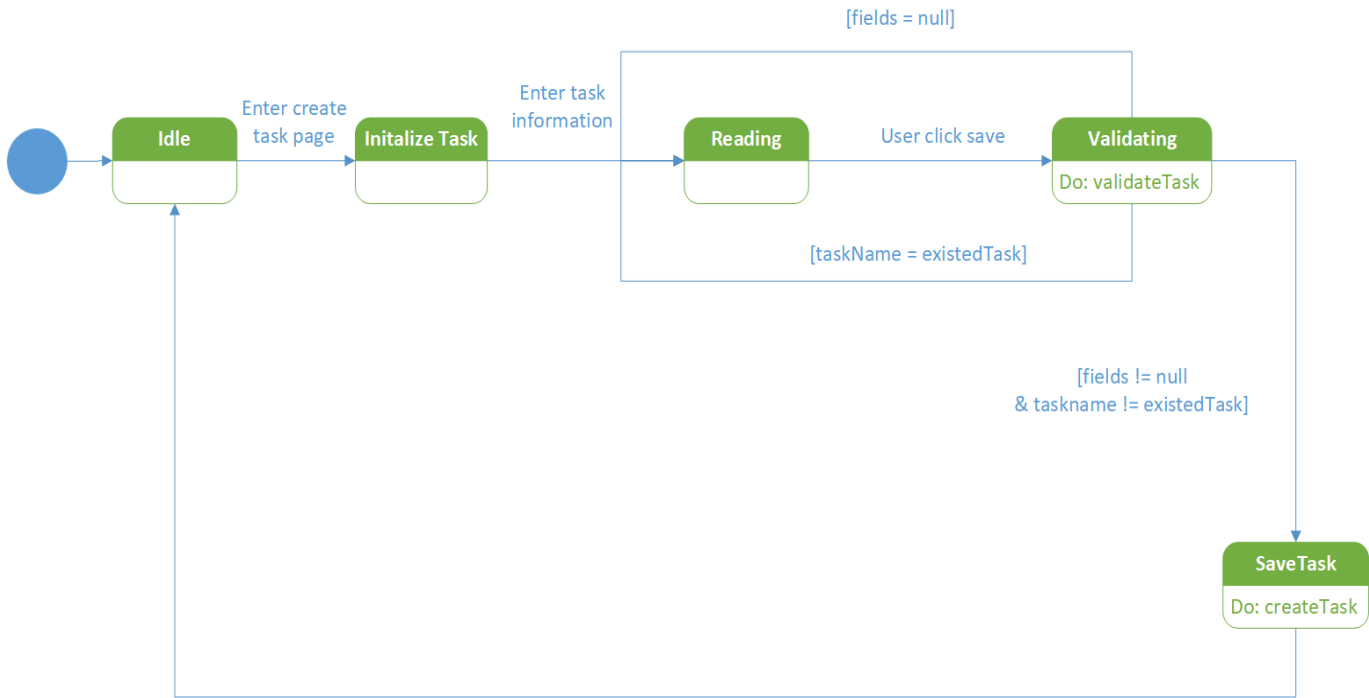
Do you have more brand guidelines (logo, design system, images)?	Marketing Officer	We have a logo, use the same color palette of the logo, but images and typography you can use your own
Do you have any specific requirements for the design and layout of the product?	Marketing Officer	The product should look simple and precise to not overwhelm the user and attract more customers.
What platform would this WebApp work on?	IT Manager	Any major browsers like Chrome, Edge, Safari, etc
What are the Functional Requirements?	IT Manager	Users should be able to log in, edit tasks and dates, mark as finished, edit preferences, and access help chat.
What are the Non-Functional Requirements?	IT Manager	Availability, Reliability, Recoverability, Security, Storage, Accessibility.
What are the User Interface Requirements?	IT Manager	Easy to operate, Quick in response, providing a simple yet consistent user interface.
What level of security do you want in the back-end database?	IT Manager	High level due to personal account information
What will be the languages for Back-end programming?	Programmer	MySQL and PHP
Do you need any external notification for the user in the app?	Programmer	Yes, we aim to have email reminder be sent to the user
What will be the languages for Front-end programming?	Programmer	HTML5, CSS3, JavaScript and jQuery libraries
Any specific requirements for the code?	Programmer	The code should look clean and well-documented

What quality standard should the product be fulfilled?	Programmer	The product should follow the guideline of the World Wide Web Consortium (W3C)
What time do you normally access the service provided by the new product?	Customer	Before and after I go to classes early mornings and in the evenings.
What is your main use of the product?	Customer	Reminding myself of important due dates.
How do you want to classify the tasks you are entering (based on the courses, dates)?	Customer	Based on the courses
How do you want to receive a due date notification?(emails or pop ups in the application)?	Customer	Through emails
Do you want to save log-in information in the application or do you want to re enter each time?	Customer	I prefer to have the option to save

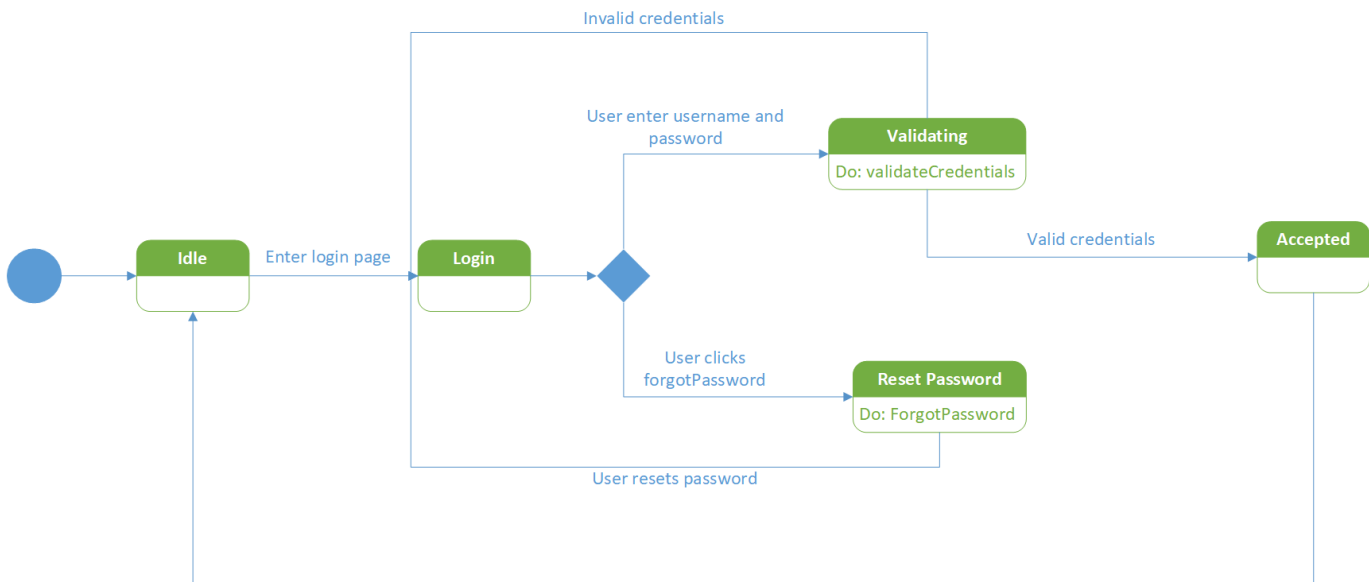
Appendix E:



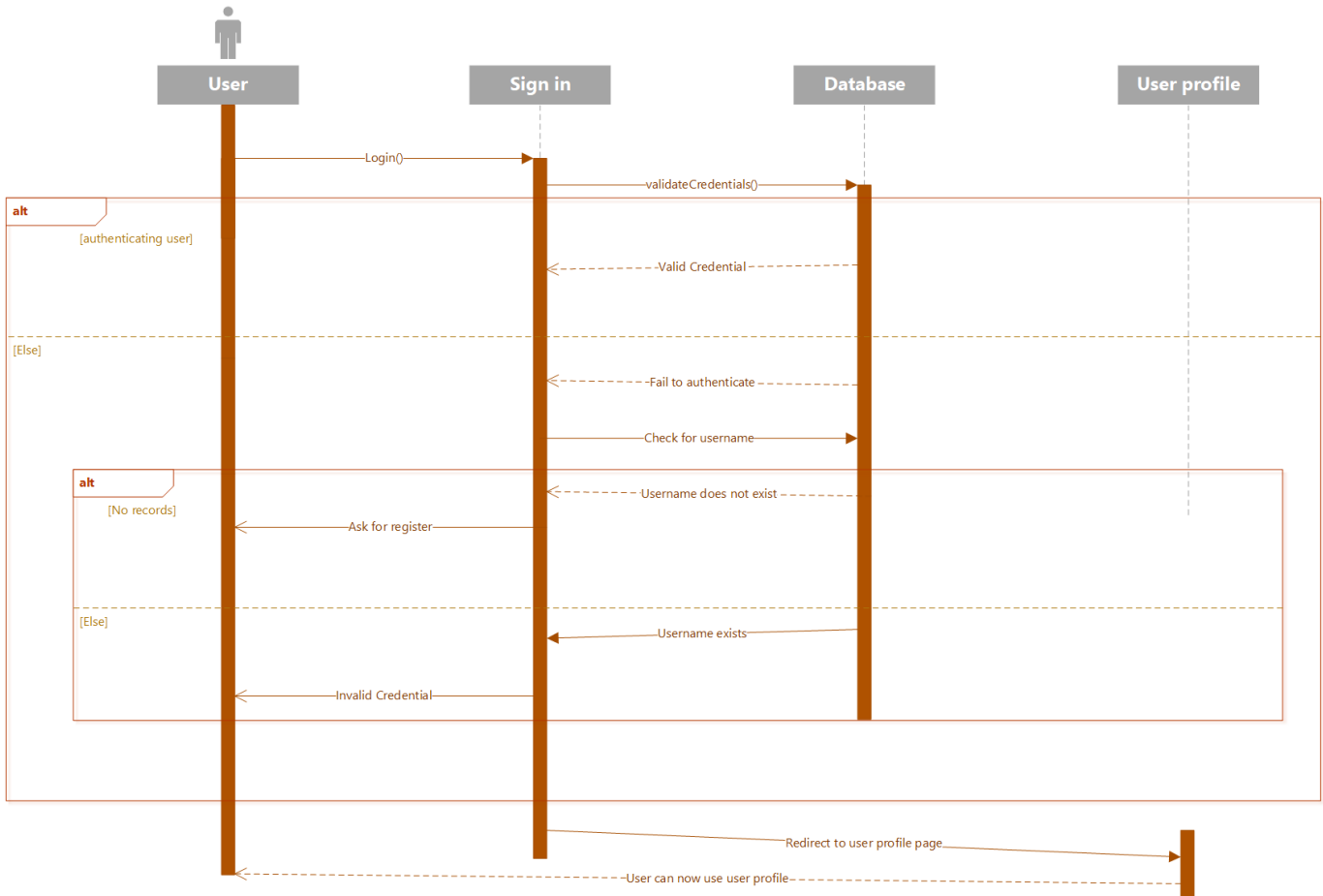
Create task state diagram



Login State diagram



Sequence Diagram:



CRC CARDS

Account	
Super Class:	
Sub Classes: User Account, Staff Account, Investor Account	
Description: Basic structure of an account	
Attributes:	
Name: User	Description
-userStatus: varchar	Account for users
-investorStatus: varchar	Account for investor
-staffStatus: varchar	Account for Staff
Responsibilities:	
Name	Collaborator
Distinguish account status	User Account, Staff Account, Investor Account
Change password	
Change email	

Login	
Super Class:	
Sub Classes:	
Description: Login functionality page	
Attributes:	
Name	Description
- Userid:int	Unique ID for identifying user

- Username:vchart	Username for login
- userpass:vchart	Password for login
Responsibilities:	
Name	Collaborator
Collect the credentials	Sign Up
Verify the credentials	Sign Up
Log the user in	Account

Sign up	
Super Class:	
Sub Classes:	
Description: Where app can be accessed	
Attributes:	
Name: User	Description
-UserID:int	Unique ID for identifying user
-Username:vchart	Username for login
- password:vchar	Password for login
- matchPassword:vchar	Confirm password
- Email:vchart	Verify email
Responsibilities:	
Name	Collaborator
Collect credentials	Account
Create Accounts	Account
Directs to log in page	Login

Investor Account	
Super Class: Account	
Sub Classes:	
Description: Investor account structure	
Attributes:	
Name	Description
-investorProfile: varchar	Account for investors
Responsibilities:	
Name	Collaborator
Check foot traffic	Login
Check the active and passive users	sign up and Log in

User Account	
Super Class: Account	
Sub Classes:	
Description: User account structure	
Attributes:	
Name	Description
- userProfile: varchar	Account for Users
Responsibilities:	
Name	Collaborator
Edit profile	Account
Create Tasks	Task

Speak at chat	Chat Box
Modify FaQ	

Staff Account	
Super Class:Account	
Sub Classes:	
Description: Investor account structure	
Attributes:	
Name: User	Description
- staffProfile: varchar	Account for staff
Responsibilities:	
Name	Collaborator
Speak at chat	Chat Box
Modify FAQ	

Customer Support Page	
Super Class:	
Sub Classes:	
Description: Page for Customer Support	
Attributes:	

Name: User	Description
Responsibilities:	
Name	Collaborator
Chat page	Chat Box
FAQ page	Staff Account

Chat box	
Super Class:	
Sub Classes:	
Description: Chat box messaging	
Attributes:	
Name: User	Description
-Message: varchar	Message customer staff
Responsibilities:	
Name	Collaborator
Manage the chat box	Customer Support page

Task	
Super Class:	
Sub Classes:	
Description: Tasks added and to be done	

Attributes:	
Name	Description
- Task IDs	Unique code of task
- Task name	Name of the task
- Task category	Category of the task
- Task due date	due date of the task
- Task status	Status of the task
Responsibilities:	
Name	Collaborator
Create Tasks	
Delete Tasks	
Modify Tasks	
Automate Tasks	Automation task management

Automation Task Management	
Super Class:	
Sub Classes:	
Description: link of external accounts updating the agenda	
Attributes:	
Name	Description
Responsibilities:	
Name	Collaborator
Link with external sources	

Populate due dates	
--------------------	--

Party Analysis Pattern

The party analysis pattern cannot be used in the context of our project. Due to its simple nature and straightforward design it is not beneficial for our use case diagram. There are only 3 main actors with well defined roles where creating a Party Analysis Pattern would prove unfruitful for its own purpose.