Qatar University

College of Engineering

Department of Computer Science and Engineering

Senior Project Report

Internship Application

Project Group Members:

Sultan Alemadi (201902749)
Nawaf Al-Sowadi (201807684)
Abdulaziz Al-Kubaisi (201610832)

Supervisor: Mohammed Saleh

2023

This project report is submitted to the Department of Computer Science and Engineering of Qatar University in partial fulfillment of the requirements of the Senior Project course.

Declaration

This report has not been submitted for any other degree at this or any other University. It is solely our work except where cited in the text or the Acknowledgements page. It describes work carried out by us for the senior project. We are aware of the university policy on plagiarism and the associated penalties and we declare that this report is the product of our own work.

Student: Sultan Alemadi Date: 5/18/2023

Signature:

Student: Nawaf Al-Sowadi Date: 5/18/2023

Signature:

Student: Abdulaziz Al-Kubaisi Date: 5/18/2023

Signature:

Abstract

In today's rapidly evolving technological landscape, we have transcended numerous boundaries, allowing machines and algorithms to perform a vast array of tasks and jobs with little to no human intervention. These advancements offer numerous benefits, including enhanced efficiency, precision, and productivity while alleviating unnecessary burdens. In this context, we have identified a pressing need for a comprehensive system to streamline and manage all aspects of the internship program at Qatar university, ensuring seamless communication among all parties involved.

Our project is dedicated to creating an innovative and user-friendly system that addresses the challenges faced during the internship process, improving the overall experience for examiners, mentors, trainees, employers, and focal points. By developing cutting-edge software, we aim to provide course coordinators with a highly accessible platform for obtaining essential information and significantly enhancing communication. Moreover, this system will automate many tedious tasks, increasing efficiency and convenience.

The Key achievements of our project include a request management function between coordinators and employers, the ability for trainees to view offers, distinct user profiles for each user type, grading capabilities, live chatting, and much more. These novel features set our design apart, making it unique and impactful for all parties involved. For trainees, this streamlined approach provides ample opportunities to showcase their skills and potential to prospective employers. By cultivating an environment that fosters personal and professional growth, our project ultimately aims to enrich the internship experience for all.

Acknowledgment

First, we would like to thank Allah for giving us the necessary power and energy that pushed us to our very limits toward success. In addition, a huge thank you to our great supervisor, Dr. Mohammad Saleh, who followed up with us weekly and suggesting the best ideas and innovations. His direction helped us shape this huge project, and we cannot go a day without thanking him for his tremendous efforts. Also, we would love to thank Eng. Mohammed Mohammed, who was like a second supervisor and offered us help many times, and we are incredibly grateful. Finally, we could not forget the other group of students with whom we merged in the middle of our project. The wonderful time we spent with them will not be forgotten. We worked, laughed, and complained, but in the end, we shared the same goal, and all six of us are working hard toward it.

Table of Contents

List of Figures

List of Tables

1. Introduction and Motivation

The significance of internship programs in today's competitive landscape cannot be overstated. Many aspiring trainees, predominantly students, eagerly seek internships to gain valuable experience. However, in Qatar, the process of finding and managing internships is notably challenging due to the absence of a centralized platform that connects all employers. Consequently, trainees and employers alike struggle with the instability of existing procedures. The internship department coordinator faces the brunt of these difficulties, grappling with the complex task of managing communications among multiple parties.

Currently, internship coordinators must face stressful processes, such as manually querying extensive databases and coordinating with large organizations. This demanding responsibility weighs heavily on their shoulders, even though the necessary technology to streamline the process is readily available. These challenges not only affect the coordinator but also impact trainees and cooperative employers.

Internships provide college graduates with essential first-hand experience, professional opportunities, and personal growth, making them more competitive in the job market. By participating in internships, students can develop relevant skills and potentially receive recommendations or job offers. However, the convoluted process within Qatar's universities adds undue stress to students and all other participants, potentially undermining the program's objectives.

Our proposed solution aims to significantly improve the internship program's workflow by automating most tasks and eliminating manual processes. This user-friendly system will enable coordinators to access critical information and communicate with all parties efficiently and effectively. Trainees will be able to apply and monitor their progress with ease, while both local and international companies will enjoy a more seamless and rewarding experience when participating in the internship process.

1.1. Problem statement

The primary issue we aim to address is the lack of effective software or automated systems to facilitate the organization of internship programs, resulting in numerous challenges for all parties involved.

First, let's consider the obstacles faced by the program coordinator. Many of their tasks, such as querying large databases, could be automated. Currently, coordinators manually search the entire database to identify eligible trainees, then send emails to selected candidates and await their responses regarding acceptance or rejection. This labor-intensive and time-consuming process is not only stressful but also prone to human errors. Once the trainee information is collected, it is sent manually to the listed companies. However, there is no efficient way for coordinators to maintain communication with trainees and companies, leaving them with limited insight into the ongoing interactions. Consequently, crucial decisions made by companies regarding trainees may not reach the coordinator, leading to a chaotic experience.

Secondly, the unclear communication within the internship program can result in poor decision-making, negatively impacting trainees' experiences. This could lead to trainees not learning valuable skills or even dropping out. A meeting with Qatar University's internship section revealed the challenges students face, such as being assigned work unrelated to their field of study. For instance, last year, some computer engineering students were tasked with computer science-related work, contrary to the program's objectives. Similarly, computer science students were given tasks that did not utilize their coding and programming abilities.

These significant issues can be resolved by developing a modern application that serves as a central hub, connecting everyone and streamlining communication. This virtual platform would include a comprehensive database and robust communication options, enabling more effective management of internship programs.

Technical challenges:

- Ensuring system reliability and continuous operation.
- Achieving scalability, as the user base is constantly expanding, necessitating the selection of an appropriate database to manage and store vast amounts of data.
- Designing an intuitive, user-friendly interface that can be quickly understood by users.
- Ensuring robust security measures to protect sensitive data and uphold client privacy.
- Developing the system to be compatible with all popular platforms, requiring careful consideration of the development foundation.

- Non-Technical Challenges:
- Persuading trainees, employers, and coordinators to adopt and utilize the proposed application.
- Encouraging coordinators and cooperative companies to consistently provide detailed requirements.

A complex engineering problem is one that can be analyzed and resolved through the application of engineering methodologies [1]. Our objective is to transform the antiquated and inefficient practices currently employed in internship processes into seamless, automated workflows that require minimal effort. By developing and implementing well-engineered algorithms, we can significantly enhance the internship experience, shifting it from a stress-inducing ordeal to a more engaging and welcoming opportunity. At present, no such solutions exist within the Qatari context. We strive to introduce our innovative approach, contingent on the collaboration and support of employers and coordinators.

1.2. Project objectives

- ❖ Design and develop a comprehensive in-house solution that seamlessly connects all parties involved in the internship process.
- ❖ Develop an intuitive and user-friendly application encompassing all essential features, catering to the needs of coordinators, trainees, and employers.
- ❖ Increase the appeal of the internship program to a broader range of companies by enhancing accessibility, ultimately providing students with more opportunities to develop their skills and gain valuable work experience.
- ❖ Minimize errors arising from manual tasks performed by coordinators and employers during the internship process.
- ❖ Ensure that the developed application is flexible and adaptable to accommodate future changes, whether related to functionality or general internship program rules.

1.3. Expected benefits and impacts on various contexts

Our project is a pioneering solution in Qatar, addressing a previously unexplored issue that holds significant implications for the future of internship programs. The impact it will create is both substantial and essential. As previously mentioned, our primary objective is to transition from manual processes to automated workflows, leading to increased efficiency and accuracy. Furthermore, we aim to streamline communication among all participants, fostering a transparent and mutually beneficial process.

The primary goal of this project is to elevate the importance of internship programs, enabling companies to identify and hire trainees who are well-suited to their requirements. Internships not only offer valuable, hands-on experience in a real-world work environment but also provide insights into potential career paths and job trajectories. Such opportunities allow participants to apply the knowledge acquired during their internships to future roles [6]. However, the demand for internships currently outpaces the supply. Among the 31.5% of students who have not completed an internship, over 70% reported unsuccessful searches — an outcome we are determined to prevent. Research indicates that graduates who complete internships at three or more sites are more likely to receive full-time job offers [7].

The success of this ambitious project could have a significant impact on the workforce. Our team chose this challenge because it pushes our boundaries and tests our capabilities to their limits. As a team of three students, addressing a substantial problem and devising an innovative solution will not only enhance our reputation but also position us as key contributors to the field.

Table 1. Expected benefits and impacts

Context	Expected benefits and impacts
Individuals	Individuals will be able to easily engage in various internship opportunities, providing them with invaluable experience that enhances their skills and adaptability in any workspace. Additionally, internships will leave a lasting impression, shaping their career development.
Organizations	An influx of competent workers, equipped with internship experience, will contribute to increased productivity and reduced onboarding burdens for companies. This ultimately leads to a more efficient and effective workforce.
Society	The overall outcome for society will significantly improve as more people participate in internship programs. With the process being more straightforward and accessible, competition will drive an increase in average skill levels among individuals.
Global	Our application enables international employers to register and search for available trainees, thanks to its accessibility. Although trainees are often sent abroad, they frequently face communication issues with their internship coordinators and assigned employers. Our solution addresses this problem by providing a reliable means of communication, bridging the gap between local trainees and global opportunities.

1.4. Market Research and Business Viability

In recent years, Qatar has experienced a significant technological leap, leading many companies to seek out a new generation of talent with immense potential and the ability to adapt to modern technology. Qatar places great importance on its education system, with numerous universities emerging annually, covering a wide range of fields. Consequently, more students are

graduating in majors that were less prominent a decade ago. While these graduates possess the necessary knowledge, the critical question remains: can they effectively apply it in the real world?

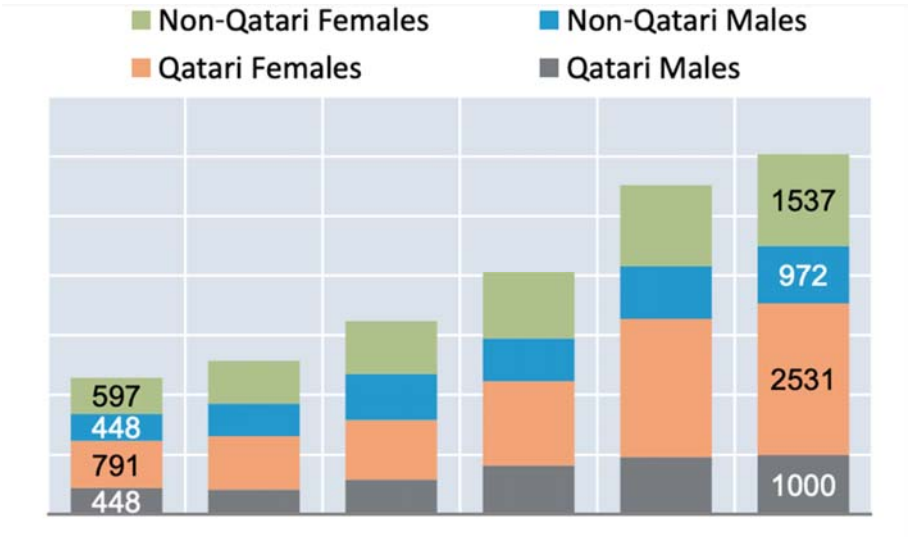


Figure 1. Number of graduated students in recent years [17]

As we can see in the figure above, the percentage of students who graduated is on a sharp and constant increase but sadly most of them do not get the chance to experience a real working environment. Often, they try to dodge questions that are related to real-life experiences in job interviews. We conducted a survey on 38 Engineering college students from Qatar university and here are the following results:

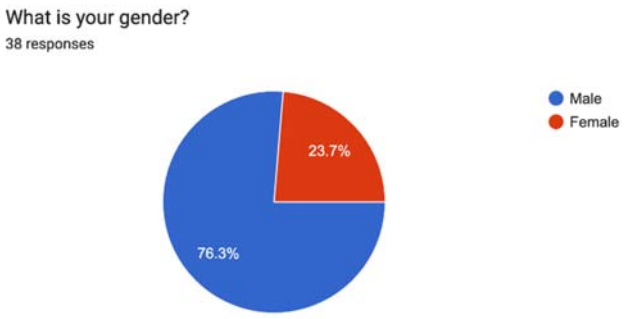


Figure 2. Applicants gender

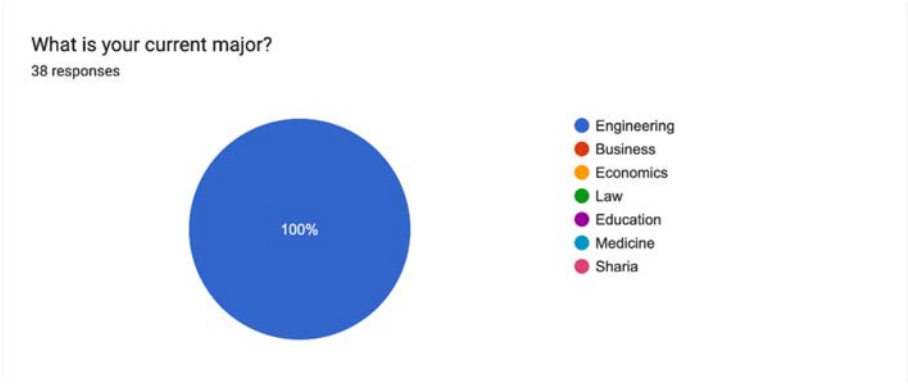


Figure 3. Current major

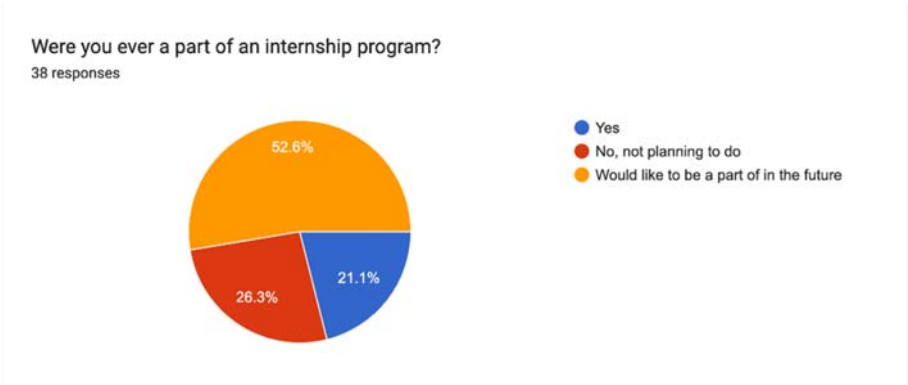


Figure 4. Being a part of an internship

52.6% of the applicants want to have a chance at an internship. This huge percentage dictates that a lot of students know the benefits of practical training, and they are planning to be a part of it. The important question is, does the current state of applying for an internship fulfill the goals of these students?

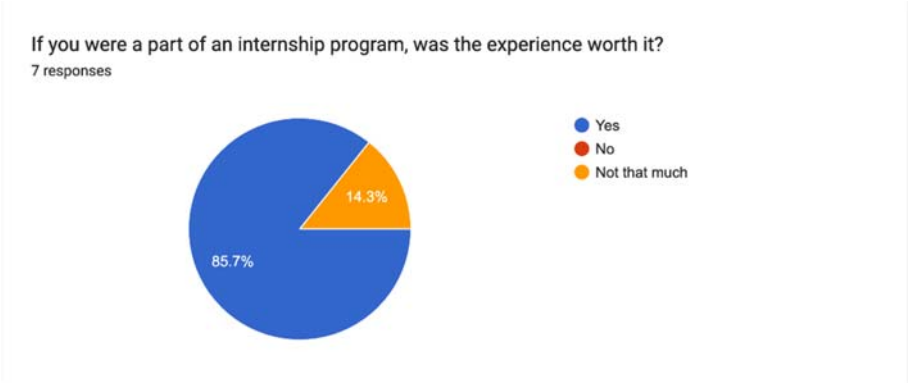


Figure 5. If the experience was worth the time

The experience of being able to participate in a practical training program is undeniably worth it. Even if it was not the smoothest experience, it has personal benefits that can only be gained by participating.

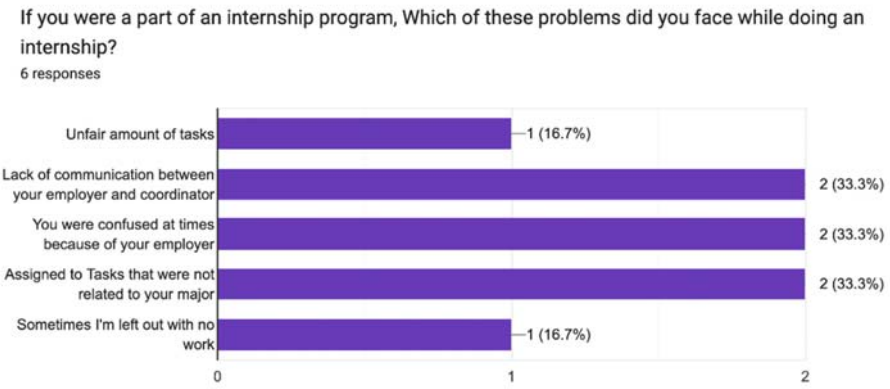


Figure 6. Problems faced in Internship.

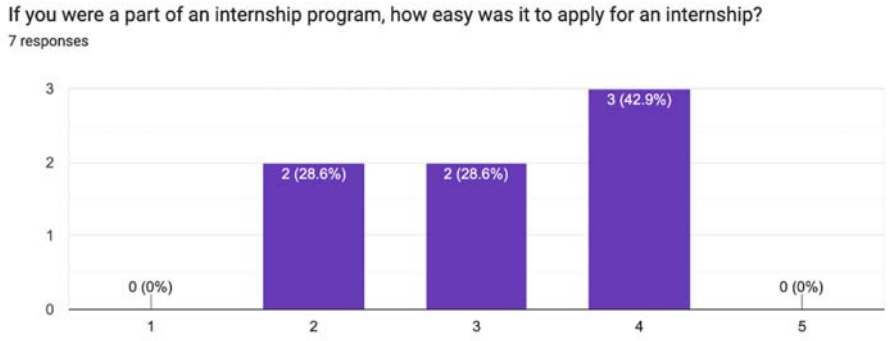


Figure 7. Difficulty of applying for an internship.

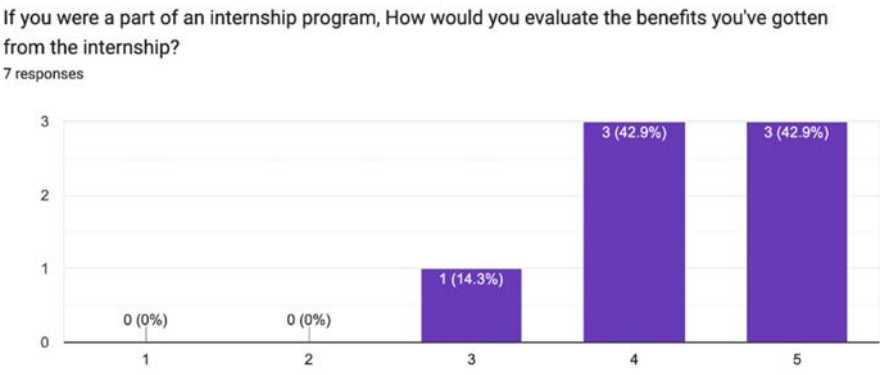


Figure 8. Internship benefits evaluation

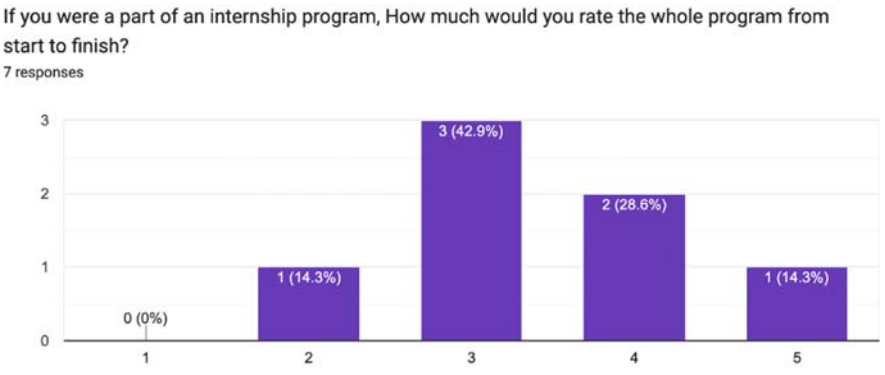


Figure 9. Rating The Program

Based upon the last four Bar Charts, we notice a pattern here that the current procedures and the way the internship program is being run, is causing stress on students. They all agreed that it is beneficial, but only a few of them agreed that the experience was as easy as it should be. Luckily, our application solves a lot of the mentioned issues due to our thought-out features and with this valuable data, we can focus more on issues students have faced. Simplifying the process of applying for an internship is our main priority because it will boost the popularity of the program.

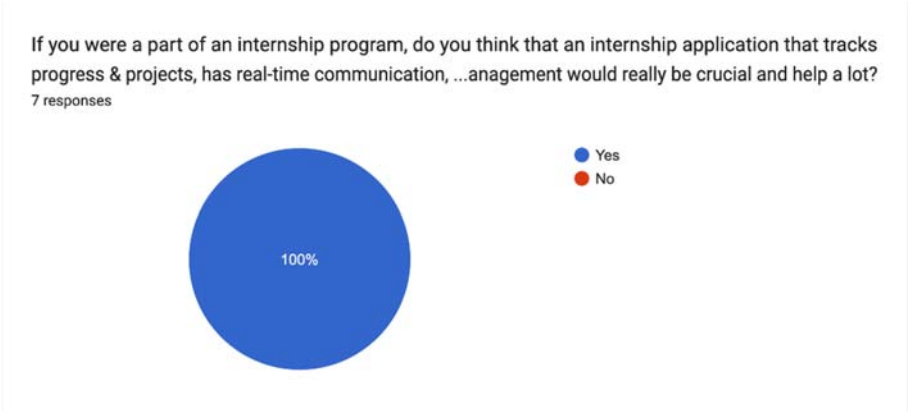


Figure 10. Importance of an internship

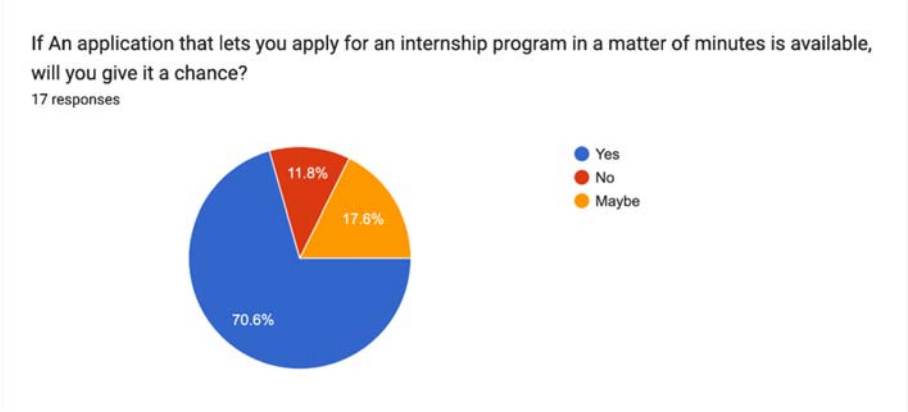


Figure 11. probability of giving our project a chance

Figure 12. Internship Key features

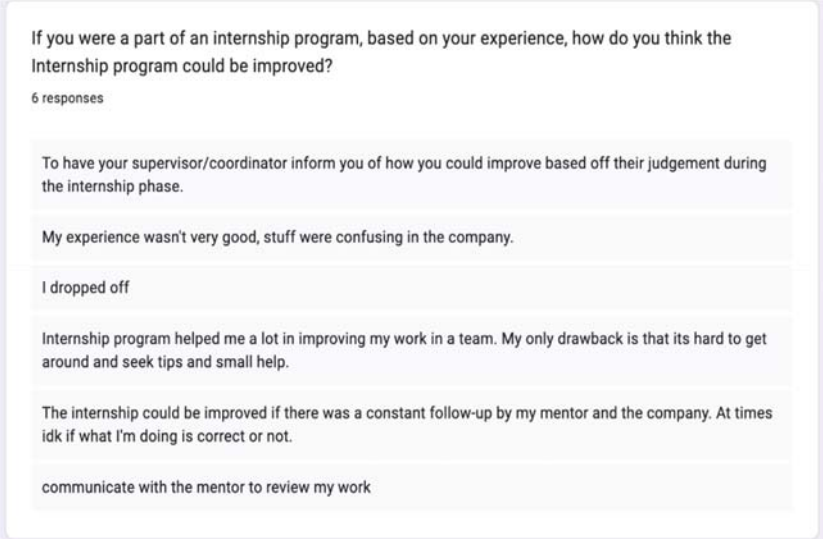


Figure 13. Applicant's suggestions

Our application is important based on the data that we have extracted from students who did an internship or are willing to do one, as it will solve most, if not all, of the mentioned issues. The young people are waiting for a chance to prove themselves. The whole internship process will be remade, and this system is going to be the first of its kind in Qatar. All the other systems and applications are international and cannot be accessed locally; they do not support usage in Qatar. To top it off, the system is going to be completely free of cost, it can be downloaded via the App Store or Google Play and the website will be compatible with mobile devices and computers.

2. Background and related work

2.1. Background

2.1.1 What's an internship?

An internship is a professional learning opportunity that offers trainees hands-on work experience related to their field of study or career interests. Internships allow students to explore and enhance their chosen professions, acquiring new skills in the process [12]. They involve a period during which trainees apply practical knowledge pertinent to their major, equipping them with essential skills for their future careers. Employers often prefer candidates with practical experience, further emphasizing the importance of internships. The challenges faced by universities in managing internships inspired us to embark on this project.

2.1.2 Who is a Trainee?

A trainee is an individual participating in an organization's training program, temporarily assuming this title while interning at a company. Trainees are typically assigned small tasks designed to test and improve their skills. Our application offers trainees various options, such as selecting

suitable offers, tracking progress, and engaging in real-time chat with mentors, providing both freedom and convenience.

2.1.3 Who is a Coordinator?

A coordinator is a person responsible for assembling various components or people to complete a project [16]. In our project, the coordinator oversees the internship, monitoring trainees' progress and providing necessary support. They also serve as the liaison between the department and the employer, initiating requests for available offers and negotiating terms. Coordinators essentially tie the entire program together.

2.1.4 Who is the employer?

An employer is an individual or organization with employees, directing their work and dictating how tasks are completed [13]. In internships, employers host trainees and designate focal points to manage their affairs. Coordinators typically initiate contact with employers before handing negotiations over to the focal point. In our case, employers quickly register through our application and appoint a focal point to represent them.

2.1.5 Who is the Focal point?

The internship focal point, also known as the coordinator, supervises the creation and execution of an internship program, managing site supervisors and overseeing the program's progress [15]. The focal point plays a crucial role in the internship, bearing responsibility for its success. In our application, the focal point receives credentials from the employer and logs into our system to provide personal information and access various functionalities, such as handling requests from coordinators, creating internship projects, and managing trainee and mentor assignments.

2.1.6 Who is the Mentor?

A mentor is an experienced professional who provides informal guidance to less experienced individuals, tailoring their mentoring style to suit the protégé's needs [14]. Mentors play a critical role in internships, working closely with trainees on tasks and offering clear directions for project progression. In our application, mentors are added by the focal point and assigned to available projects, primarily responsible for submitting evaluations based on trainee performance.

2.1.7 Who is the Examiner?

The examiner is responsible for assessing a candidate's performance in exams or other evaluation forms. In our application, the examiner reviews final rubrics, reports, and presentations, then submits and uploads grades.

2.1.8 What is a department?

A department is a distinct functional area within an institution, with its own organizational structure [19]. The department selects the internship program coordinator and plays a limited role in the program. In our system, the department registers, and hands over the user to the coordinator, who continues the process.

2.2. Related work

2.2.1: riipen.com

The main goal of Riipen is to give students experiences so they can graduate and are prepared for the workforce. Students who are immersed in industry projects acquire skills necessary for the workforce. With actual business ventures, Riipen brings together business and academia. Projects might be finished as remote internships or as part of the program itself. Their technology enables the ecosystem's participants to jointly develop, oversee, and evaluate large-scale authentic experiential learning opportunities. In North America, Riipen is the top technological platform for any school wishing to offer micro-experiential learning. Figure 14 shows the user interface with the options Riipen provides. For example, Riipen subscribers will have the option to have a profile, feedback, projects, and schools. In figure 15, a dashboard screen for Riipen is illustrated. This dashboard screen provides the users a state of the account, activities, and projects. Figure 16 shows the search feature that Riipen has to search companies and apply for their internship. Figure 17 displays the bookmark for companies' internship feature that Riipen has. Another feature that Riipen has is chatting as in figure 18. Currently, there are 75,000 students (about the seating capacity of the Los Angeles Memorial Coliseum), 10,000 employer partners, and more than 300 academic institutions in their market [1].

The issue and the main tradeoff with the Riipen are that the service is limited to some countries. For example, when searching for an internship in Qatar, there is no single match. Also, it is so difficult to have access to the application. Our university tried to have access to the application, but they said there was no response.

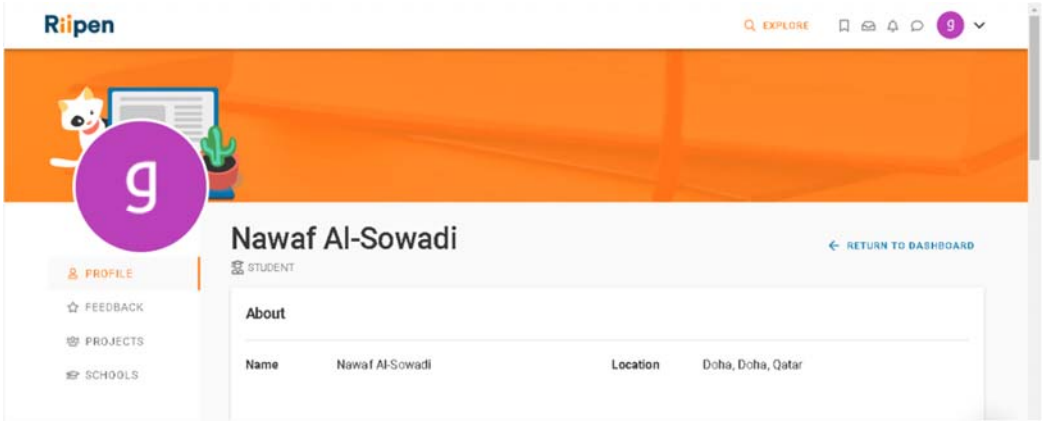


Figure 14. UI Riipen

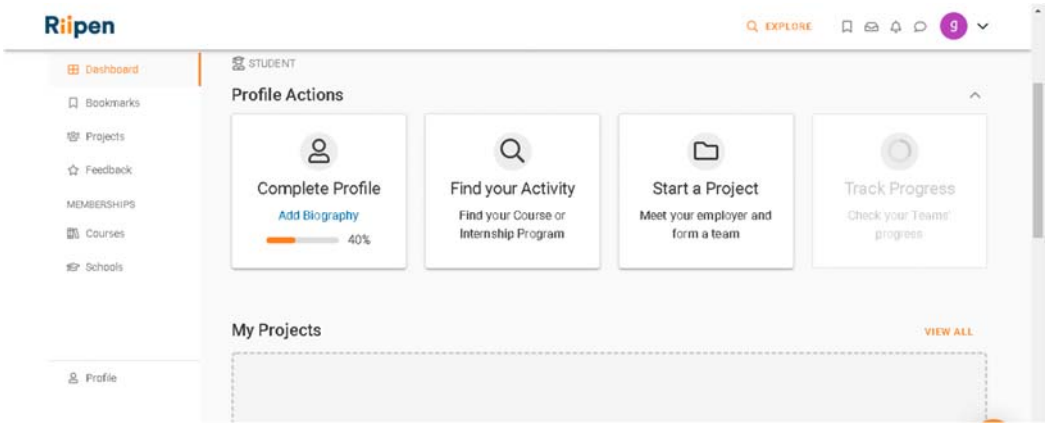


Figure 15. Riipen dashboard

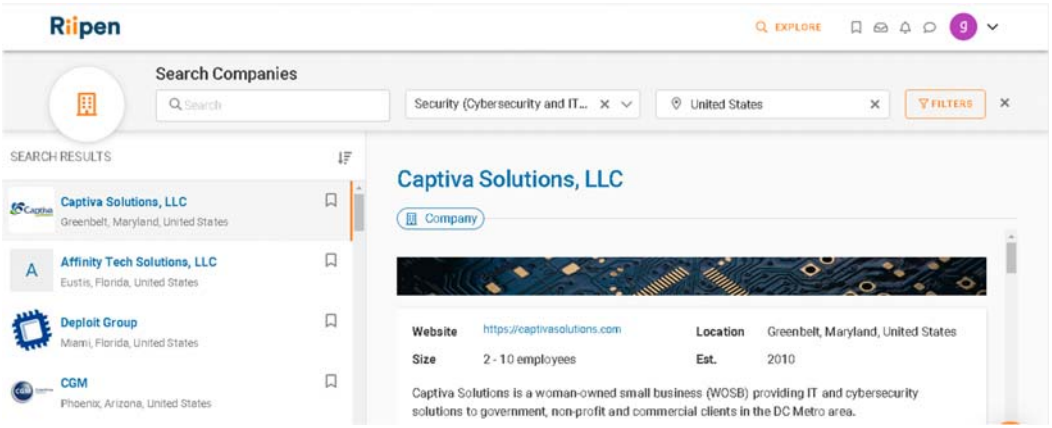


Figure 16. Riipen search

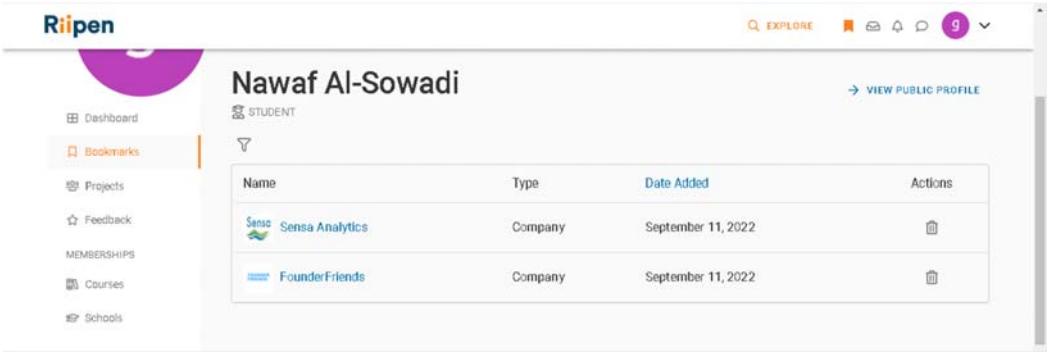


Figure 17. Riipen bookmarks

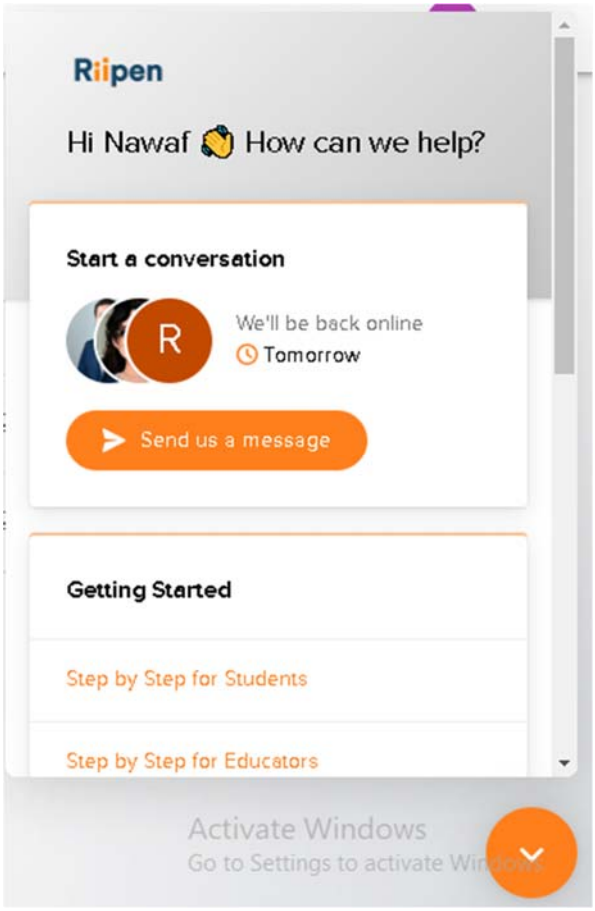


Figure 18. Riipen communication

2.2.2: Internshala.com

A reputable Indian company is Internshala. Therefore, your chances of landing a job greatly increase if you complete any training or an internship following your internship. They provide many educational options, internships, and training. Anyone who needs it can choose between training and an internship. Programming, personal development, and digital marketing are just a few of the courses that anyone can take to easily advance their talents. Any course from here can be taken for a very low price. And signing up is simple [2]. Figure 19 shows the user interface that Internshala has. This user interface provides the internships the website provides. Searching for internships with filters to meet the requirements of the intern is a feature in Internshala as in figure 20. When we went through the website, we faced an error in the bookmark as in figure 21. We as a programmer should not have same error.

The main issue with this application is that it is limited to only one country. Which means that all the companies and internships are in India. Also, the application has some errors during the bookmark process. In addition, the design looks old and not that good; it could be better.

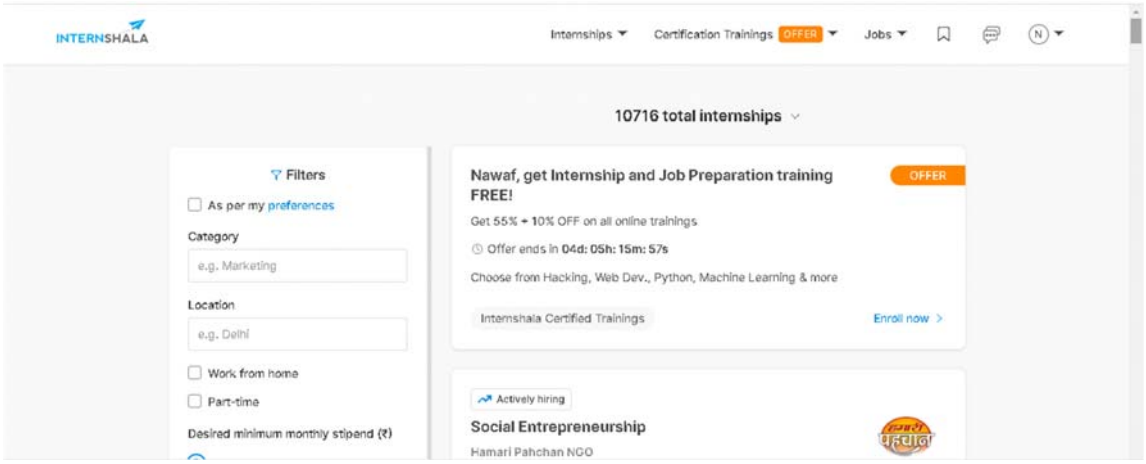


Figure 19. Internshala UI

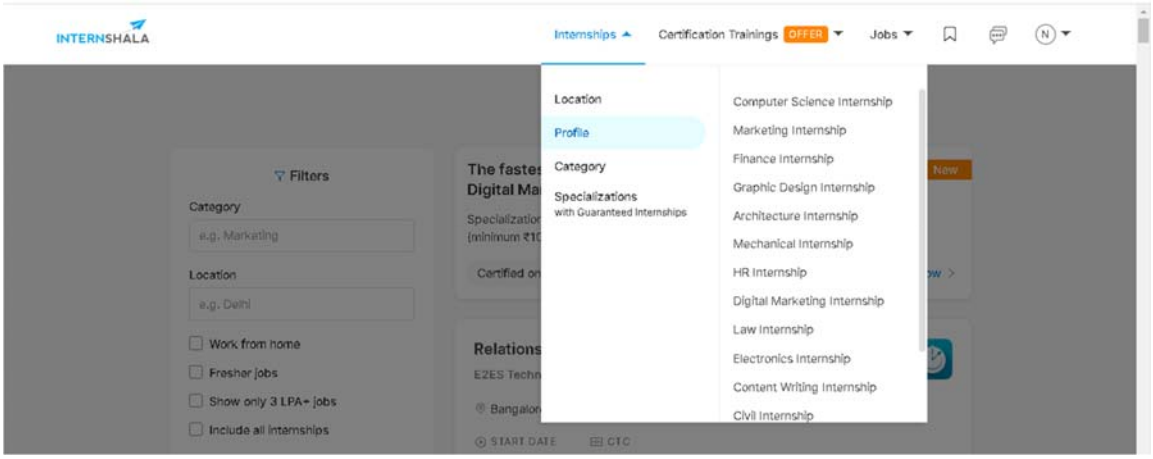


Figure 20. Internshala search options

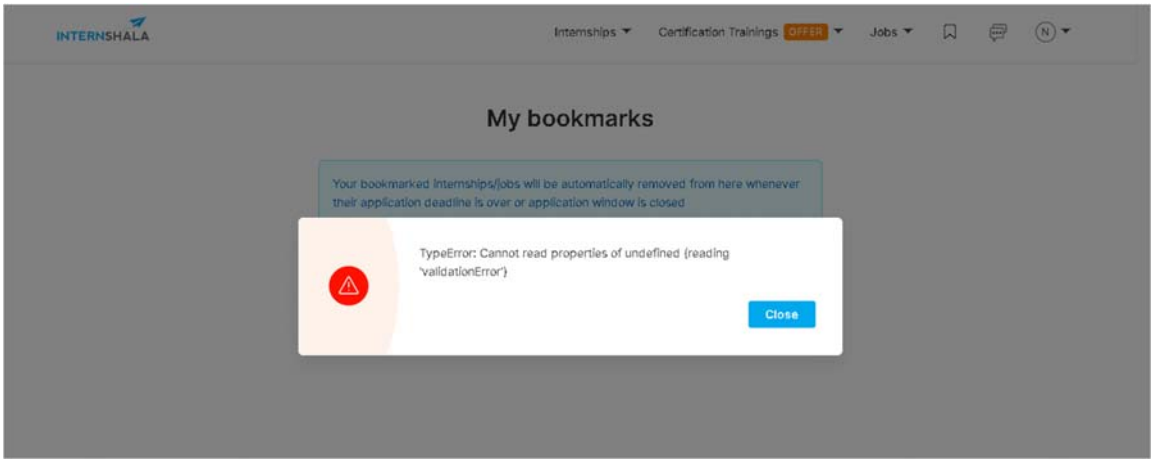


Figure 21. Internshala bookmark problem

2.2.3: Naukrigulf.com

Employers may find top talent on Naukrigulf.com, and job seekers can use it to find their ideal position. It is a forum designed to close the information gap between companies and job searchers, allowing them to do so swiftly, efficiently, and affordably. It was established as the Gulf region's fastest-emerging job portal since its launch in early 2006. Figure 22 shows the home page of the Naukrigulf with a nice design and a lot of options. Naukrigulf provides a dashboard as in figure 23 to summarize all of the activities and the necessary options for the user. As in the previous related works, Naukrigulf provide search option to search for internships as in figure 24. Figure 25 illustrates how the communication between the employers and the trainee is established. The site receives daily visits from thousands of job seekers from the UAE, Saudi Arabia, Bahrain, Kuwait, Oman, Qatar, etc. [3].

The problem with this app is that it does not support internships. In addition, the app doesn't have two important features which exist in the previous apps which are bookmark and request status. Also, its design is not simple and hard to use, especially for beginners.

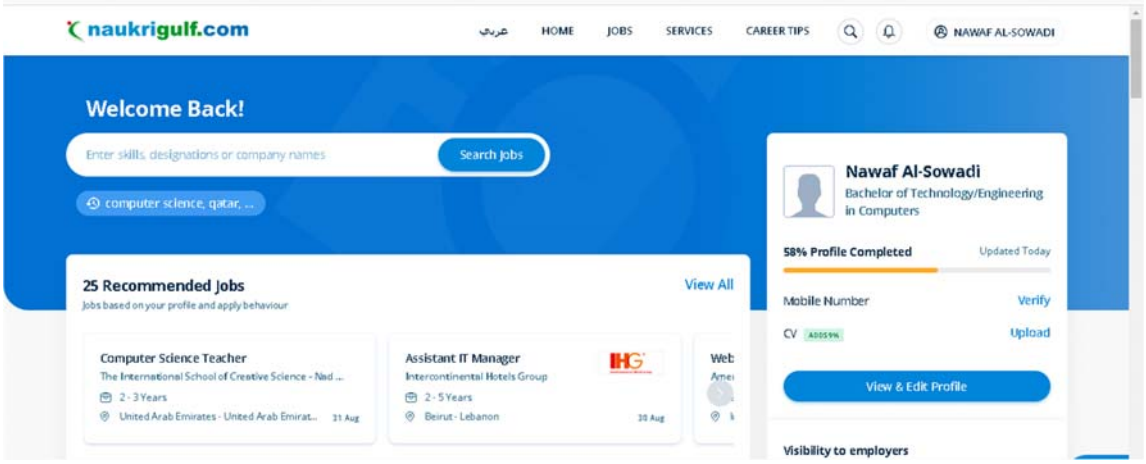


Figure 22. Naukrigulf UI

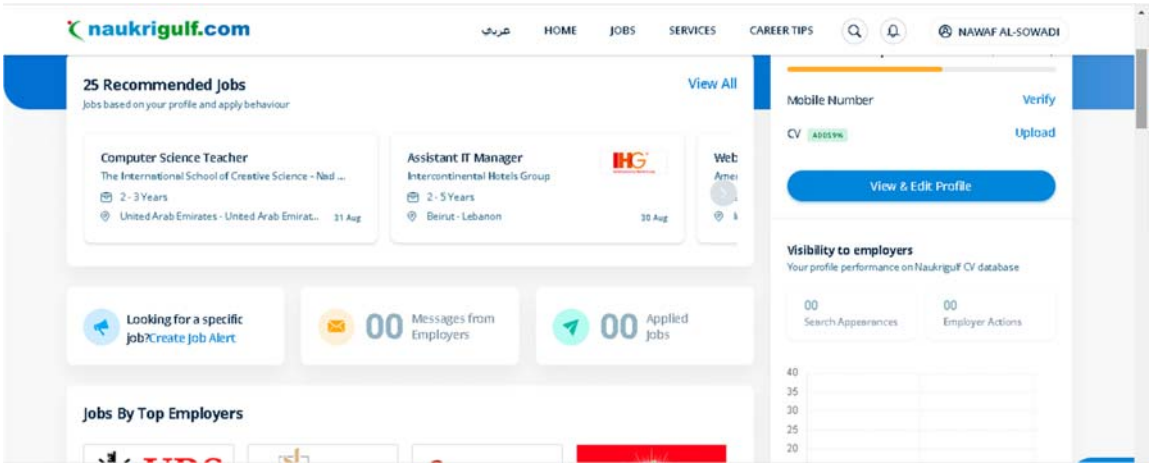


Figure 23. Naukrigulf dashboard

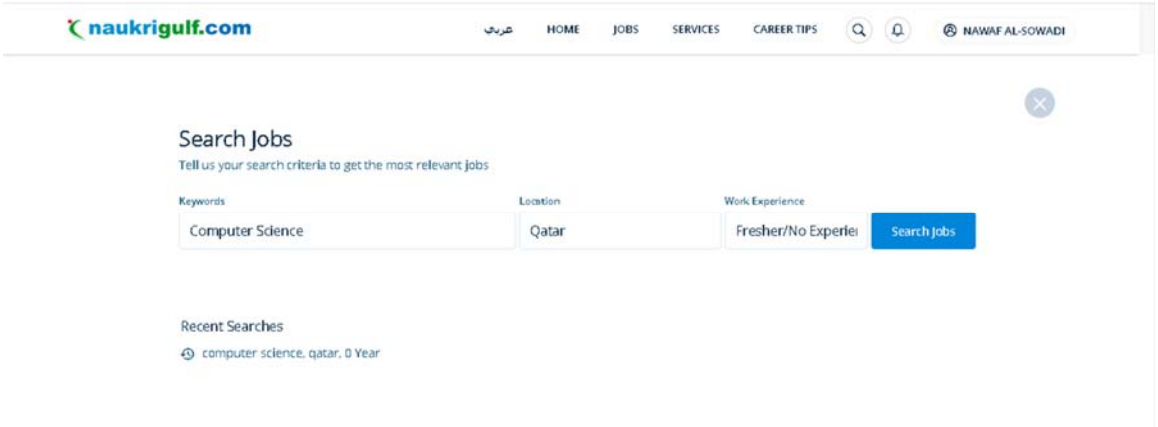


Figure 24. Naukrigulf search

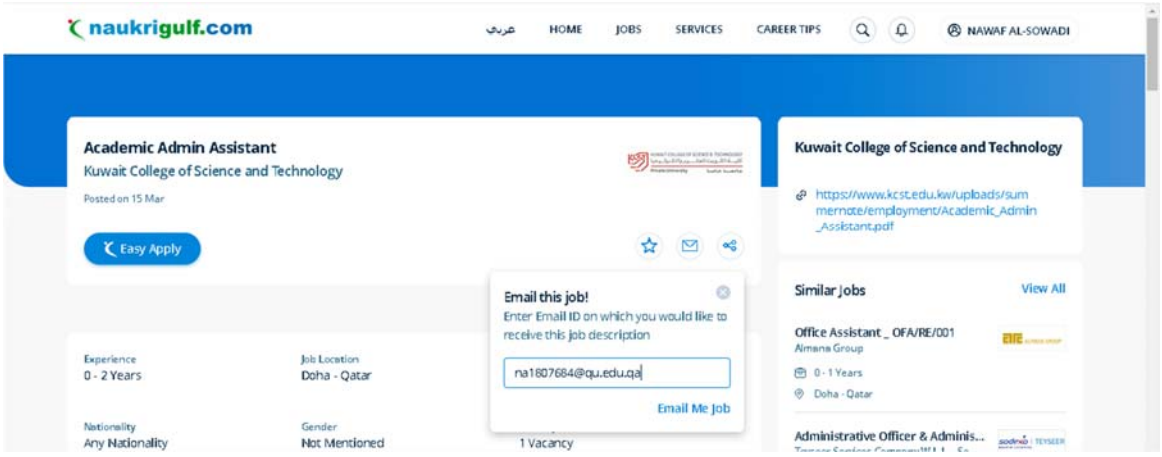


Figure 25. Naukrigulf communication

2.2.4: iaeste.org

IAESTE A.s.b.l., also known as the International Association for the Exchange of Students for Technical Experience, is a non-profit organization made up of national committees that represent the interests of academia, business, and students. Through career-focused professional internships abroad, social and intercultural reception programs, international networking, and other career and employer branding activities in more than 80 countries globally, it serves 3500+ students, 3000 companies, and 1000 academic institutions. IAESTE provides a registration for the student as in figure 26. Figure 27 illustrates the success of the student registration where the website sends the credentials to the student email. In addition, figure 28 displays the polices of the website. The home page in figure 29 displays a lot of options for the user like news, helping with technical issues, …etc. In figures 30, 31, and 32, one of the most important things in the internships is the offers with a filter option, internship nominations, and feedback reports. The screen in figure 33 illustrates the available internships for the student. Also, a deep looking for the internship nominations and feedback reports is displayed in figures 34 and 35.

One of the issues with this application is that it is missing some features that are needed in educational internships between universities and companies. Other than that, the application is perfect.

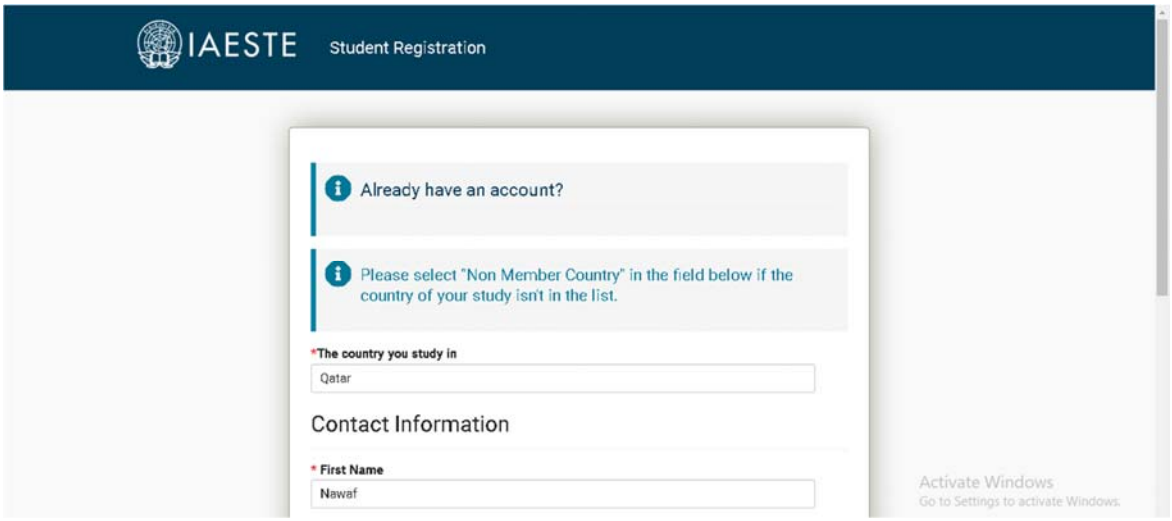


Figure 26. IAESTE Student Registration

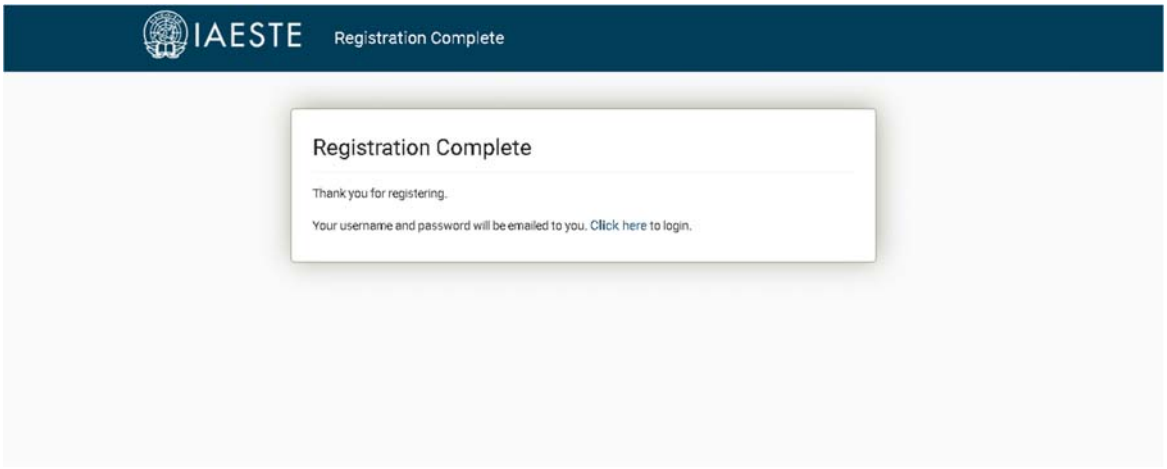


Figure 27. IAESTE Sent Credentials After Registration

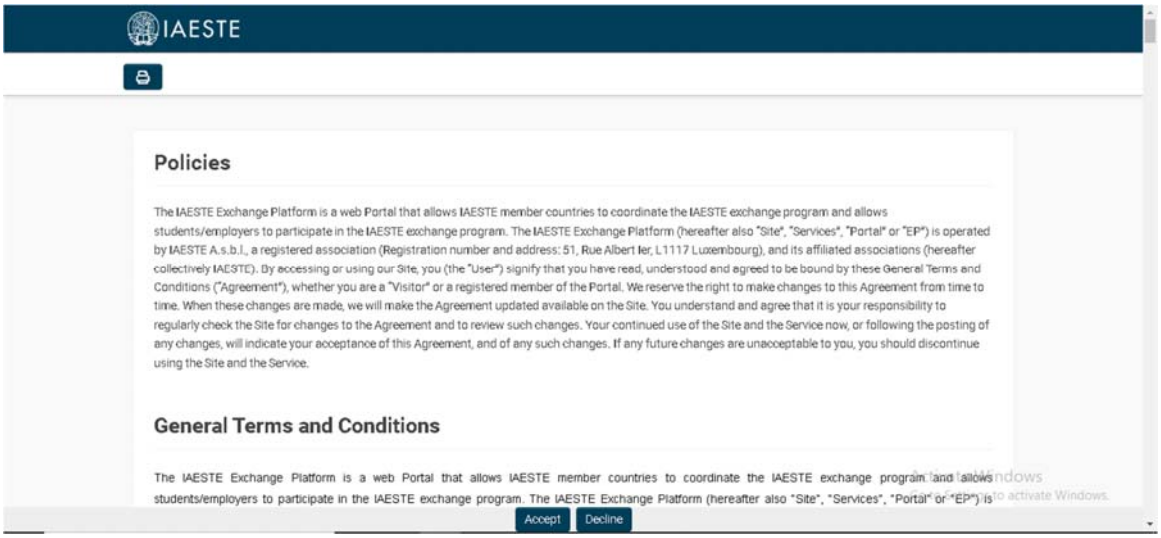


Figure 28. IAESTE Policies

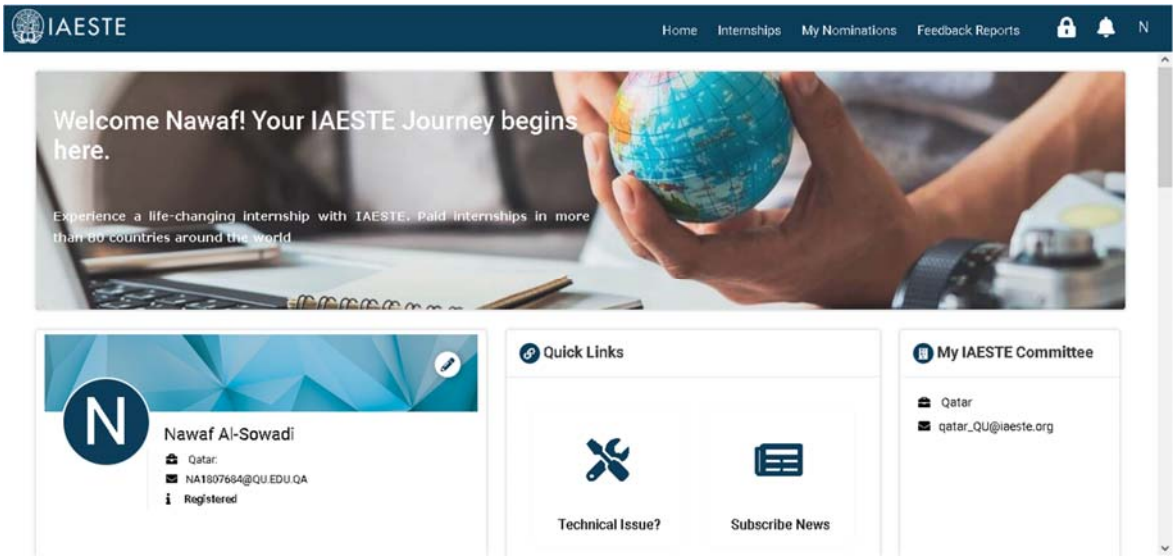


Figure 29. IAESTE Home page

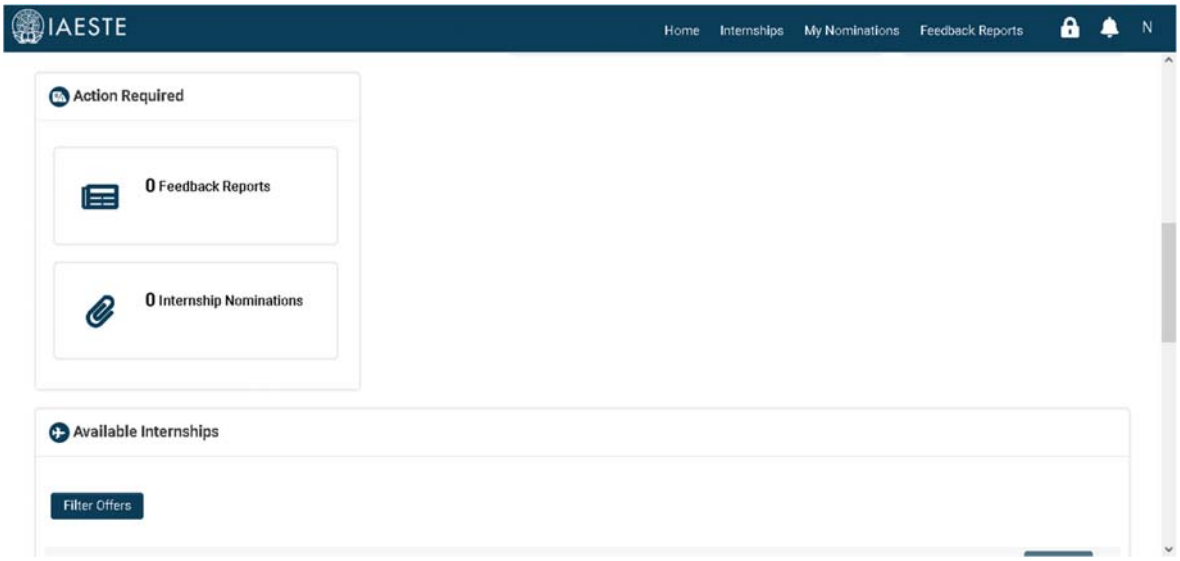


Figure 30. IAESTE Internship page.

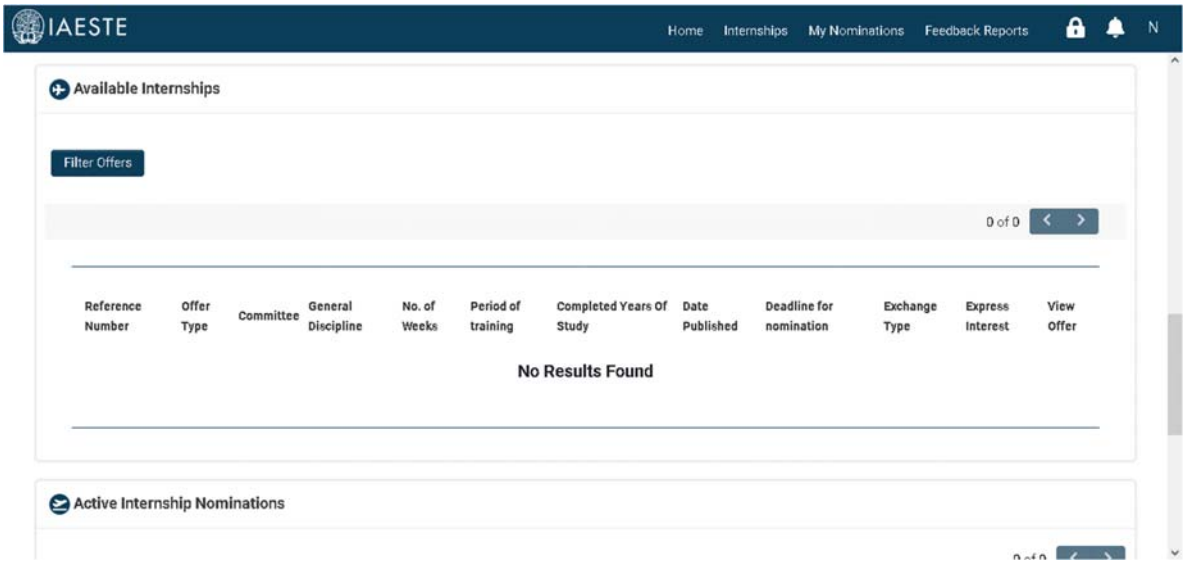


Figure 31. IAESTE Offers Page.

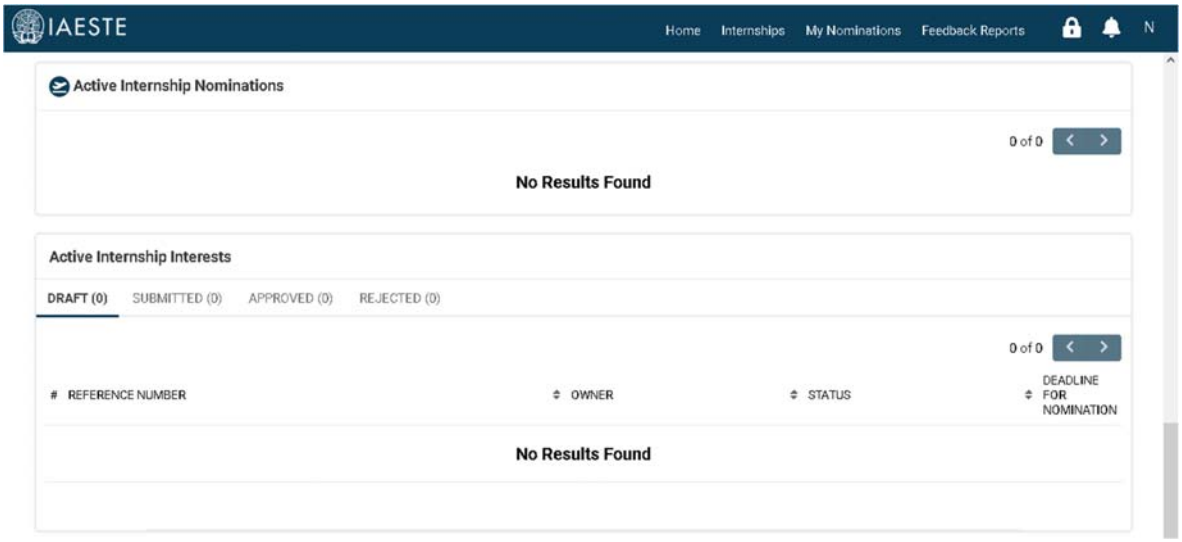


Figure 32. IAESTE Offers Page. (2)

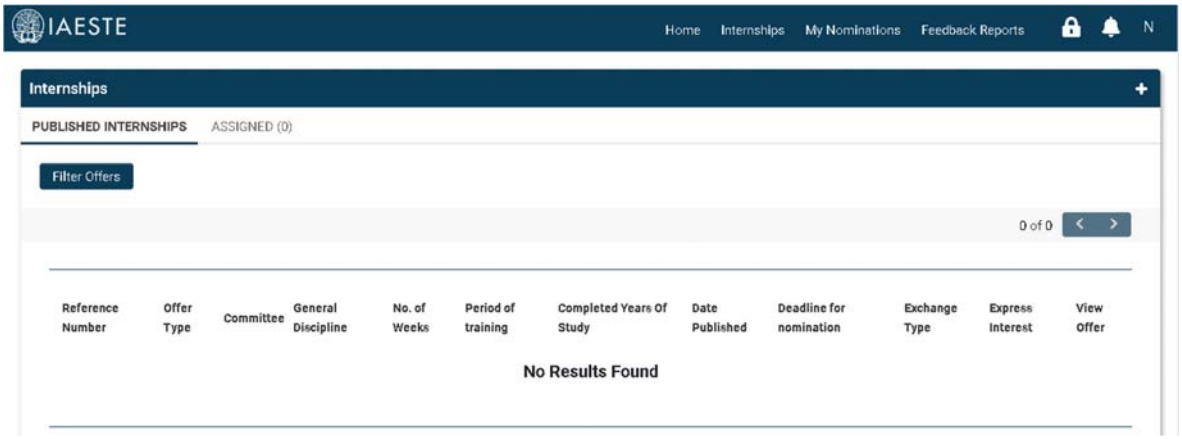


Figure 33. IAESTE Internship Page. (2)

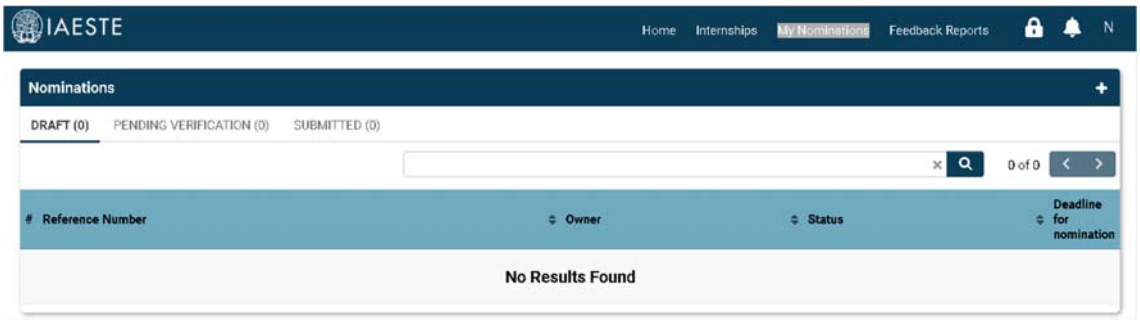


Figure 34. IAESTE Nominations Page

Figure 35. IAESTE Feedback Reports Page

2.2.5: Microsoft Teams

Microsoft Teams in figure 36 is an all-in-one hub for "workplace communications," "collaborative cooperation," "video chats," and document sharing. Furthermore, Microsoft Teams is a useful e-learning platform and the most popular online learning tool that can be utilized for both online and mixed-mode learning [20]. Very few Employers and coordinators use Microsoft teams' application to run an internship program. It is not a popular method to lean on while managing an internship because it was not designed to handle it in the first place. It is just a good application to hold meetings on, share files, chat. But that is basically the high limit of Microsoft Teams. It does not have enough capabilities to match the other applications we have mentioned previously.

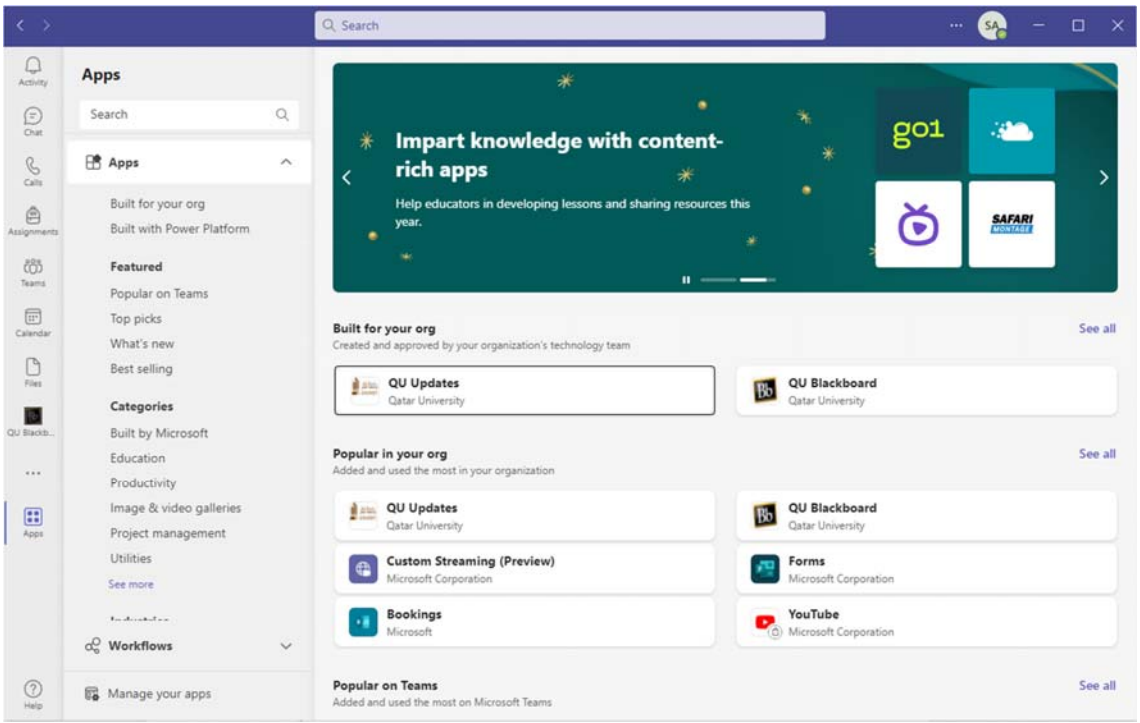


Figure 36. Microsoft teams

After discussing some of the related work, we would like to make a comparison between most of them and our solution. This comparison is discussed in table 2.

Table 2. Related Works Comparison Table

Features/Related Works	riipen	internshala	naukrigulf	IAESTE	Our solution
Dashboard	Profile actions, My Projects, Feedbacks as cards, and tables respectively	-	New Recommended Jobs, Message from employers, applied jobs, Jobs by Top Employers, Career Tips, Employer's View of My Profile, Featured Employers, Featured Consultants	Action Required, Available Internships, Active Internship Nominations, Active Internship Interests	A dashboard that includes all the necessary options to simplify the usage of our application. Moreover, it will be like a hub where you will be able to navigate to your needs.
Filtering Internships based on different factors	Category, Location	Location, Profile, Category, Specializations	Skills, Designations, Location, Work Experience	Country, General Discipline, Internship Type, Reference Number:	Filtering will be done based on the internship program provided by the employer.
Bookmark internship	Easy and convenient bookmark	Simple bookmark	-	Table of available internship	Ideal bookmark
Communication	Very nice chatting	Simple chatting	Email-based	Email-based	A full one-to-one live chatting function.
Requests status	Table	Weird design - Could be better	-	Table	Table contain the needed info to track progress.

2.2.6: Key features & differences

To put it briefly, our application will significantly enhance the internship process by efficiently catering to a diverse range of users, including trainees, coordinators, focal points, mentors, and more. With a user-friendly dashboard customized for each user type, we aim to deliver a smart and intuitive experience. For instance, coordinators will have access to filtered applications and an interactive tracking system, while an easy-to-use interface will facilitate swift communication between trainees and potential employers. Additionally, new employers will be able to register with ease, further simplifying the process.

3. Requirements analysis

3.1. Software development process

Various software development processes are available for consideration, including the waterfall model, prototype model, V-model, Agile methodologies, and others. After thoroughly examining our project and conducting an in-depth analysis, we have determined that a hybrid approach, combining the waterfall and prototype models, would be the optimal choice for our software development process.

Traditionally, software development projects have followed a structured, process-driven approach. However, the emergence of Agile methodologies has introduced a more adaptive strategy for software and systems development [8]. Our chosen methodology combines the waterfall model with a prototype, as it offers numerous advantages that are well-suited to our project that will be discussed in 3.1.1.

Figure 37 illustrates the 7 phases of the combined model. The first phase involves identifying system requirements, as detailed in Appendix A. This is followed by the software requirements phase, where each requirement is thoroughly discussed in section 6.1. These requirements are determined based on the project's objectives. Once the requirements are identified (due to the model's nature), the analysis phase commences to evaluate the requirements and document the outcomes, which can be found throughout the report. Next, the program design phase begins, where we create use case diagrams, sequence diagrams, class diagrams and Figma designs that we built upon its ideas, to provide a comprehensive overview of the system. Completing these phases allows us to construct a prototype.

The coding phase follows the design phase, where we implement the requirements analysis and design. In the first phase, we developed 30% of the system, with the remaining development to be completed in Senior 2. Before deployment, the system undergoes testing to ensure error-free operation. The final stage is operation and maintenance, where the software is made publicly available, and any bugs are addressed.

The first four steps are included in the prototype construction process, enabling us to revisit and revise them upon completing each milestone. This flexibility allows for changes and modifications to ideas and requirements before initiating the coding phase. This aspect of the software development process ensures thorough evaluation and analysis of each step, given their somewhat loosely coupled nature.

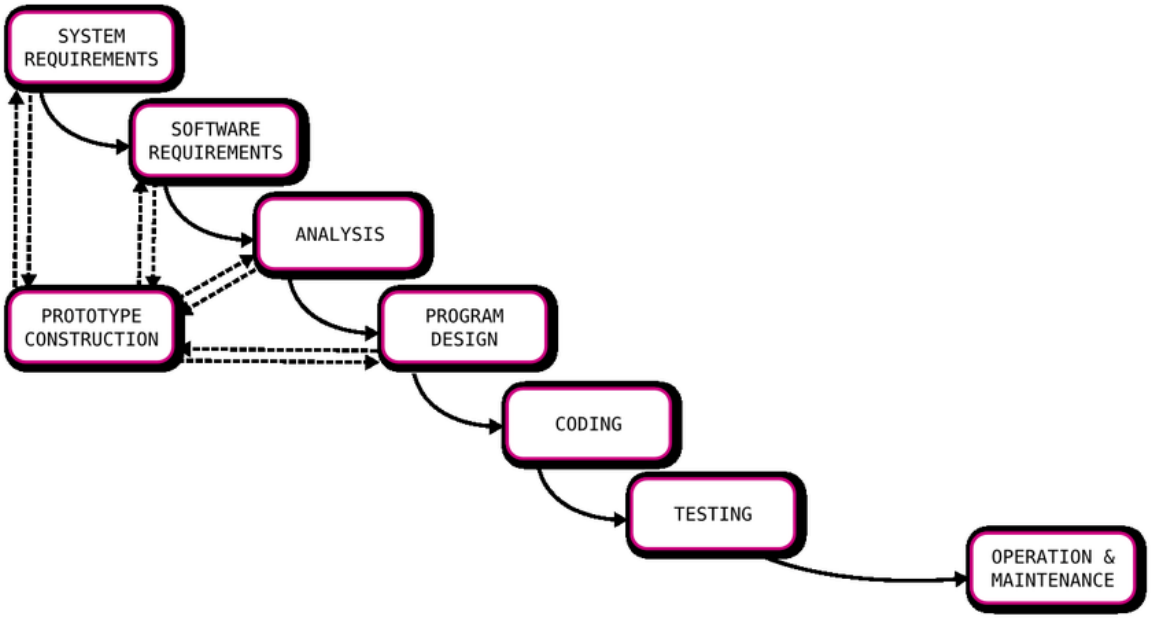


Figure 37. Waterfall and Prototype development process

3.1.1. Applying the software development process

We chose the combination of waterfall and prototype models because it offers numerous advantages that are well-suited to our project. This model encourages best practices, such as defining requirements before design and designing before coding. It is also easy to understand and implement, with a systematic and disciplined approach that identifies deliverables and milestones. Furthermore, it is document-driven, making it suitable for both large, mature projects and small teams.

The combined model enhances user usability, delivering a system that closely aligns with users' actual needs. It also improves design quality, maintainability, and reduces development tasks. Crucially, this approach lets us develop a prototype based on the analysis of system and software requirements, and program design. As it involves constructing a prototype this will enhance the result of each stage and ensure the best quality and no-error results before starting to implement the actual software in stage 5.

It is true that this software development process has many advantages, but it involves extensive steps. These steps are used to ensure the quality of the application but sometimes during the development of our software these steps were not necessary, and it took up a lot of time that we could use it in other aspects. But overall, it helped us a lot to develop our application in a professional way, and it helped us to ensure the best quality.

3.2. Functional requirements

The following use case diagram that is figure 38 was created based upon our ideas as a team and with the suggestions we have received from the meetings we have conducted with the head of the internship program in Qatar University. We have listened to their issues and wishes that we will try to fully fulfill. As detailed in our software development process section, this use case diagram represents the culmination of our system requirements analysis. These requirements have been studied and refined over time to align with the users' needs. The use case diagram encapsulates all these insights and the data gathered, offering a comprehensive overview of the solution we aim to develop.

Our focus here is to improve communication between all users who are involved. There will be a quick registration process for various types of users. Also, a comprehensive dashboard that will guide the client and direct him. And most importantly as we mentioned is establishing communication between any two users will be straightforward and easy to understand. Moreover, some small features such as changing the theme of the app, changing password, customizing profiles and more to come.

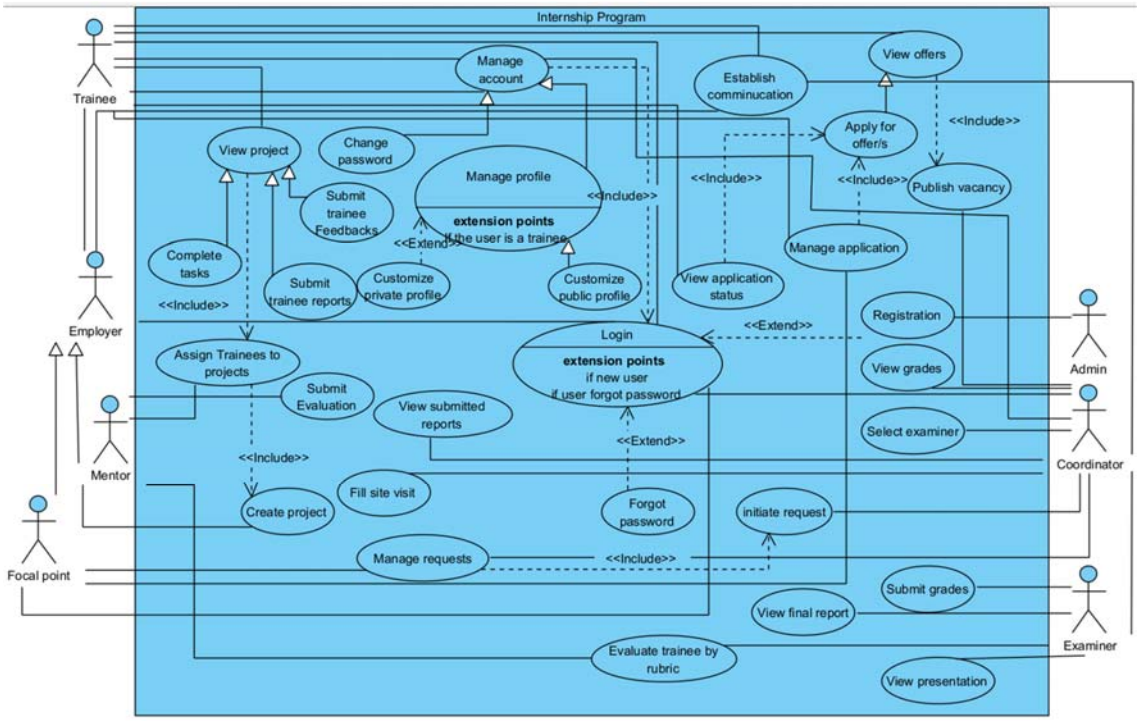


Figure 38. Use cases diagram.

Table 3. Use cases summary.

Use case	Brief description
UC**01.** Login	Requires credentials to log into the system.
UC**02.** Registration	Handles the registration of accounts.
UC**03.** Forgot Password	The system will send an email to the user followed by multiple verification steps to set a new password.
UC**04.** Initiate Request	Communication will be established with a selected group of Employers by the coordinator.
UC**05.** Manage Request	Responsible for handling the negotiations of requests between the focal point and the coordinator.
UC**06.** Publish Vacancy	Coordinator creates a vacancy for a specific employer.
UC**07.** View Offers	Display offers from various employers.
UC**08.** Apply for Offers	Submits trainee's applications to the employers.
UC**09.** Assign trainee to project	Specify a project for a trainee.
UC**10.** Create a project	Mentor creates a project for the trainee to be assigned.
UC**11.** View project	Trainee can see his assigned projects.
UC**12.** Complete task	Trainee can complete his tasks in his projects.
UC**13.** Submit trainee reports	Trainee submits his reports to the system.
UC**14**. Submit trainee feedback	Trainee submits his feedback to the system.
UC**15.** Submit evaluation	Mentors submits his evaluation about the trainee.
UC**16.** Manage Application	Manage the state of the trainee’s application to a specific offer.
UC**17.** View Grades	Displays a trainee’s grade to the coordinator.
UC**18.** Establish Communication	Communication between two parties will be established.
UC**19.** Select Examiner	The coordinator will be able to select an examiner and assign students to the examiner.
UC**20.** Submit Grades	A trainee's grade will be submitted by the examiner.
UC**21.** Fill site visit	The coordinator will fill out the site visit.
UC**22.** Manage profile	Display the profile Information with the ability to customize the profile.

UC**23.** Customize private profile	The user will be able to customize his private profile.
UC**24.** Customize public profile	Display all the available settings.
UC**25.** Change password	change the password of the account.
UC**26.** Manage Account	Display the account menu of selections that the user can enter any of the options like manage profile, change settings and view application status.
UC**27.** View application status	Trainee can view the application status of the projects he applied for.
UC**28.** View submitted reports	This use case is for the coordinator to view the submitted reports by the trainees.
UC**29.** View final report	The examiner will view the final report that was submitted by the trainees.
UC**30.** View presentation	The examiner will view the presentation submitted by the trainees.
UC**31.** Evaluate trainee by rubric	The examiner and mentor will view the evaluation rubric to evaluate the trainee.

3.3. Non-functional requirements

Table 4 presents a detailed evaluation plan for seven non-functional requirements of a proposed system, including availability, security, usability, scalability, portability, performance, and privacy. It outlines the success criteria for each requirement and the associated plan to evaluate these criteria. The evaluation strategies range from performance testing and security measures to user surveys and legal compliance checks. The table provides a clear framework for ensuring that the system meets essential user expectations and industry standards.

Table 4. Evaluation plan for the non-functional requirement

Non-functional requirement	Success Criteria	Evaluation Plan
Availability	The system must be running 99% of the time for all types of users. Only in maintenance mode where it can go offline, and it should not take that much of a time.	We will be testing the availability of our system by performing a lot of read and write actions in a short span of time. And we will increase the number of connections and messages until we reach a breaking point.

Security	Our system, handling sensitive data, employs robust security measures. These protocols are not only strong but also adaptable, ensuring long-term data protection amidst evolving cyber threats.	Our backend service provides strong security measures to protect data. By default, Firebase applies security rules that require authentication and validation of data access. These rules can be customized. Also, Firebase authentication is a service that is provided by Firebase. It stores passwords as hashes and prevent authorized access. Additionally, we will perform the brute-force attack method on our system. It is one of the most popular attacks and can serve a good indication on our security levels.
Usability	The user interface design must be simple and user-friendly. It should be straightforward and without the need for prior knowledge. All users should be familiar with it from the first usage.	Conduct a survey that evaluates the ease of use of the application. The survey will be distributed to over 30 different users. And their feedback will be taken seriously.
Scalability	The system must handle a lot of users because it's the core nature of our application. A lot of concurrent users should be expected, and the application should be ready to run with no failure.	Locust, an open-source load testing tool is used to simulate various user scenarios and measure the scalability of the application under different loads. A lot of VMs will be launched at once to test the limit of concurrent users and all of them will perform actions in the application.
Portability	The system will be implemented as both a mobile application and a web application. This will increase the user base of the application by making it easy to access.	We will be testing the functionality of the application on Android, iOS and windows systems and solve any error that occurs.
Performance	Providing a smooth performance is a main goal that we must achieve. The system should be fluid and efficient in its usage to satisfy its users. All processes should have a maximum of 5 seconds to perform.	Calculate the average response time related to each action. And if the performance of an action exceeds the desired goal, we try to modify it by using the proper algorithm.

Privacy	No private information will be shared by the system without the owner's permission. Any further information will not be gathered without consent either.	As they are required by law, we shall abide by the legal rules of privacy. The user must be made aware of how the data is used and why the system requires it. The system should also inform the user whether a third party is doing the actual data processing [18]. The system will include a straightforward privacy policy that addresses users how confidential data will be managed.

3.4. Assumptions

This section outlines the various assumptions made in this project, providing an insight into potential areas of risk. These assumptions, ranging from user accessibility and understanding to application and database operations, are integral to the project's functionality, and thus need careful management and monitoring.

- The user has access to a good internet connection. Some of the application's functionality may need very stable internet connections.
- The users are expected to easily understand how to use the app, despite the fact we try to make the interface simple and easy to use.
- The web and mobile application are available 24/7 to the users.
- The database is constantly updating the information it holds.

3.5. Ethics

Table 5 showcases some key ethics and responsibilities that software engineers should follow, based on the official guidelines from ACM and IEEE. It's like a rulebook for our work. Each rule is explained in a way that shows us how to apply it in real-world situations. This includes important things like respecting others' work, avoiding causing harm, and making sure our software is tested thoroughly before it reaches users. Understanding and following these rules helps us do our job ethically and responsibly.

Table 5. Project-related Software Engineering Code of Ethics and Professional Responsibilities

From the Code of Ethics and Professional Conduct (ACM) or (IEEE):	Usage and practice to address an identified ethical issue
ACM - 1.1 Contribute to society and to human well-being, acknowledging that all people are stakeholders in computing.	Programmers must contribute significantly to the development of systems that have meaningful solutions to limit the negative effects on society; Therefore, the systems developed must have very high standards.

ACM - 1.2 Avoid harm.	Programmers should avoid any harm that could affect others. The systems designed should avoid any loss of user information as this can be harmful to them; Therefore, programmers and system designers must minimize and limit any risk of harming users due to "coding errors" and "weak security".
AMC - 1.5 Respect the work required to produce new ideas, inventions, creative works, and computing artifacts.	Programmers should be aware of giving credit for intellectual property. This is a mandatory task for programmers, as no one person should be allowed to take credit for someone else's work, even if there is no copyright to protect their work.
AMC - 1.6 Respect privacy.	Programmers must maintain and respect user privacy. Programmers must be able to develop systems that protect the user's private information from any unauthorized access to it.
IEEE - 1.01. Accept full responsibility for their own work.	The software developer should be aware of full responsibility for their own work. If any of their designed works result in errors, the software developer must accept full responsibility for their work and begin modifying it before harmful consequences occur for users of the software.
IEEE - 3.03. Identify, define and address ethical, economic, cultural, legal and environmental issues related to work projects.	The software developer must identify and report any economic, cultural, and legal issues related to their work. The software developer should report any of these issues to their managers before proceeding with their work.
IEEE - 3.08. Ensure that specifications for software on which they work have been well documented, satisfy the users' requirements and have the appropriate approvals.	The software developer must ensure that their work satisfies the user's requirements. The software developer should always be in contact with their managers to request approval step by step. If the user does not accept approval, the software developer must modify the software until the user agrees to the work.
IEEE - 3.10. Ensure adequate testing, debugging, and review of software and related documents on which they work.	The software developer must thoroughly test, debug, and review their software before publishing it to users. The software developer should perform the appropriate tests for their job to ensure that their software does not have harmful consequences in the future.

4. Project Plan

4.1. Project milestones

Table 6 provides a detailed insight of the project's journey, marking the milestones to completion. Each milestone outlines the activities undertaken, the challenges addressed, and the deliverables achieved. This systematic and progressive documentation not only highlights the project's evolution, but also serves as an analytical tool to assess the project's trajectory and performance, thereby facilitating future project planning and management.

Table 6. Milestone of the project

Milestone	Description	Deliverables
Software development process: System Requirements		
First Milestone: Brainstorming ideas	After deciding upon the topic of our project we had a lot of questions about the implementation. We started looking around for similar applications to grasp some ideas and to have a better understanding of the concept. Also, we conducted multiple meetings with various people who are linked to such topics such as the Internship advisor at Qatar's university. Those meetings were very fruitful as we had the first time the chance to talk directly to an expert on this matter.	- The team members grasp the core idea of the project. - Prepare for the next milestone.
Second Milestone: Filling Chapter 1 and 2 in the report paper.	Chapter 1 in the report is very important because it is where we show and display our knowledge about the project, and it is where we finally agree on the problem and its solutions and our objectives. Chapter 2 helped us in expanding our boundaries of ideas and thoughts into a new level. Because after visiting and signing up in those sites we had the chance to fully distinguish what are good and bad design choices to come up with the best solution.	- Fully finish the first two chapters. - Agree upon the choices we as a team made.
Third Milestone: Designing the use-case Diagram.	The use-case diagram took a lot of time and effort to finally accomplish. Too many ideas were scrapped, and others were added. A lot of meetings happened between us and our supervisor where hot discussions were held regarding some decisions and with it, we had more creative ideas. 3 weeks was the duration that we finally finished the use-case diagram with a staggering over 30 use-cases.	- Finalize the use-case diagram. - Writing down all the use-case specifications.
Software development process: Software Requirements		
Forth Milestone: Filling chapter 3 and 4 in the report paper.	After tackling the main task in chapter 3 which was the use-case diagram. The rest was simple and easy. The software development process we chose fitted the team perfectly, and the ethics were agreed upon.	- Developing a software development process.

	As for chapter four. A lot of research has gone into finding the best high-level architecture. There were too many good options such as MVC for example. But in the end MVVM was picked from the bunch because it fitted our project the most. Also, because of our previous experiences, deciding on the used hardware and software was easy to figure out. In Addition, our main solution "Flutter/Dart" was a good choice because of the many benefits that it came up with.	- Checking out the IEEE/ACM code of conduct and choosing the suitable options. - Agree on a solution and compare it to other solutions. - Agree on the high-level architecture. - Planning out which software to use in the project.
Software development process: Coding		
Fifth Milestone: Implementation	Our goal was to code as much as possible in a short time span and I believe that we have succeeded. We managed to finish a lot of the use-cases in a short amount of time in a totally new environment that we had never been into before. All members are capable to code in Dart language now and that was achieved in 3 weeks. Each week we distribute a fair share of work to each member, and the deadline is always on Saturday. The first week of implementation was creating all different registrations for all account types and their profiles. The second week was refining what we did in the first week by improving the design and logic surrounding it. Next was working on the bottom bar navigation, policies, fixing bugs before tackling the biggest use-case which is related to offers. Some contribution was made but it is not fully ready as of now.	- Registration for all account types. - Some users can register unique account types such as a mentor. - The UI theme was done. - Login is fully functional for all users. - Application is connected to Firebase database. - Some effort was done in the "Offers" section.
Sixth Milestone: Finishing the senior one report	After the implementation phase is done, we returned to the report to fill in what is left.	- Finishing the report paper.
Seventh Milestone: Full Implementation of "Offers" logic	After finishing the first senior report, we took a small break before going back to continue where we left off. In the first month we fully focused on completing everything that is related to the businesses logic of "Offers" use-cases. The first step being an initiate request from the coordinator until the last step that is accepting or denying a trainee.	-Initiate request -Publish Vacancy -Apply for an offer. -Ask for permission to view private information. - Manage Request.
Eighth Milestone: Chatting and assigning mentors/examiners.	Chatting is one of the most important solutions to the communication problem the internship suffers from. We have developed a modern and clean live chatting functionality. The employer has the option to assign a mentor to a specific offer. Meanwhile, the coordinator can assign an examiner to a trainee.	- Live Chatting. - Assign a mentor to an offer. - Assign an examiner to a trainee.

Ninth Milestone: Trainees can view their project. Focusing on Mentors and Examiners.	Develop a view where trainee can view his tasks and project details. Tunning in the experience of registering a mentor, and adding functionalities such as creating projects, adding tasks, assigning trainees and viewing their profiles. As for Examiners, he or she can view the trainees they are assigned for and view their uploaded reports or PowerPoints. In Addition, the ability to grade them.	- Create project. - View project. - Adding tasks. - Assigning trainees to tasks. - Download trainees reports or PowerPoints and grade them.
Tenth Milestone: Coordinators create tasks.	The coordinator can create tasks and submit them to all their trainees.	- Coordinator creates tasks.
Eleventh Milestone: Implement all forms/feedbacks into the system.	The objective of this milestone is to incorporate all necessary forms and feedback into the system. This includes evaluating forms, feedback, and other forms of data entry that are relevant to the project.	- Implement all given forms.
Twelfth Milestone: Working on finishing the whole report.	Shifting the focus from the project's code into the report. Chapter 5 and 7 were distributed to the whole team to get it done in a good timespan.	- Start working on the new diagrams such as sequence diagram, class diagram, activity and database. - Finish the report.
Software development process: Testing		
Final Milestone: Testing and finalizing the whole project.	Every deliverable should be 100% done before the deadline. A thorough check will happen to each one of them to ensure completion. Multiple types of testing such as integration and acceptance will be done to confirm the validity of the application.	- Examine our project to assure completion before the deadline.

4.2. Project timeline

Table 7 delineates the project timeline, outlining each milestone and corresponding tasks, and allocating them to the responsible team members. It provides a comprehensive guide to the project's execution, from initial brainstorming to final delivery. The table enables effective work scheduling and time management, helping the team to stay organized and on track. It also emphasizes the importance of continuous evaluation and documentation throughout the project's life cycle to ensure timely completion and quality outcomes.

Table 7.Project Timeline

Milestones	Tasks	Done by
First Milestone: Brainstorming ideas	Thinking about a topic	Sultan Alemadi Nawaf Al-Sowadi Abdulaziz Al-Kubaisi
	Gathering ideas and understanding our project	
	Meeting with Qatar's University internship coordinator	
Second Milestone: Filling Chapter 1 and 2 in the report paper.	Writing chapter one	Sultan Alemadi
	Writing chapter two – Background	Abdulaziz Al-Kubaisi
	Writing chapter two – Related work	Nawaf Al-Sowadi
Third Milestone: Designing the use-case Diagram.	Designing The first use-case diagram	Sultan Alemadi Nawaf Al-Sowadi Abdulaziz Al-Kubaisi
	Discussing the use-case diagram with our supervisor	
	Improving and continuing upon our use-case diagram	
	Filling out the use-case specifications	
Forth Milestone: Filling chapter 3 and 4 in the report paper.	1. Writing Chapter three – Software development process 2. Writing Chapter four – Alternative solutions and tradeoffs 3. Writing Chapter four – Hardware – Software to be used	Nawaf Al-Sowadi
	1. Writing Chapter three – Assumptions 2. Writing Chapter three – Ethics	Abdulaziz Al-Kubaisi
	1. Writing Chapter three – Non-functional requirements 2. Writing Chapter four – Selected solution overview 3. Writing Chapter four – High level architecture	Sultan Alemadi
Fifth Milestone: Implementation	1. Creating Log-in 2. Implementing the registration 3. Designing the UI 4. Creating Trainee's profile 5. Simple navigation between the screens	Nawaf Al-Sowadi

	6. Integrate Firebase database with flutter	
	1. Creating Mentor's profile 2. Add an update information function to profiles. 3. Do a bottom-bar navigation. 4. Develop a new way of updating information in all profiles.	Sultan Alemadi
	1. Fixing image selector and other bugs	Abdulaziz Al-Kubaisi
Sixth Milestone: Finishing the report	Finishing the whole report.	Nawaf Al-Sowadi Sultan Alemadi Abdulaziz Al-Kubaisi
Seventh Milestone: Full Implementation of "Offers" logic	1. Only display the trainees who applied for the specific offer. 2. Functionality to sort, filter and search trainees. 3. Add a paginator with a maximum of 5 per page. 4. Employers ask trainees for access to their private profile. 5. Trainees can either deny or approve access to their private Information. 6. Accepting or denying trainees to the offer.	Sultan Alemadi
	1. Improve the UI	Abdulaziz Al-KubaisiU
Eighth Milestone: Chatting and assigning mentors/examiners.	1. Develop the chatting UI. 2. Create a functional chatting system. 3. Modify users who can chat with each other. 4. Link the chatting to the database.	Nawaf Al-Sowadi
	1. An Employer can register and assign a mentor to an offer.	Sultan Alemadi

	2. A Coordinator can register and assign a mentor to an offer.	
	5. Develop a notifications system.	Abdulaziz Al-Kubaisi
Ninth Milestone: Trainees can view their project. Focusing on Mentors and Examiners.	1. Develop a UI for the project view. 2. A mentor can create a project based on the offer. 3. Tasks can be added with a specific deadline. 4. Assign a specific trainee to a project.	Nawaf Al-Sowadi
	1. An extra registration step to both mentor and examiner accounts type. 2. Examiner can view trainees whom he is assigned to. 3. Trainee can view his tasks in his project view. 4. The examiner can download a trainee's report or presentation.	Sultan Alemadi
Tenth Milestone: Coordinators create tasks	1. The coordinator can create a task that will be submitted to all of his trainees. Various types of tasks can be selected	Nawaf Al-Sowadi
Eleventh Milestone: Implement all forms/feedbacks into the system.	1. Implement the trainee evaluation by a mentor	Sultan Alemadi
	1. Implement an attendance form done by the mentor.	Nawaf Al-Sowadi
	1. Simplify the coordinator's UI to have better reachability and usage. 2. Implement the rubric.	Abdulaziz Al-Kubaisi
Twelfth Milestone: Working on finishing the whole report.	1. Work on chapter 5.3, create sequence diagrams. 2. Fill out all the chapters from the previous report.	Sultan Alemadi Abdulaziz Al-Kubaisi

	3. Modify certain parts to match the current state of the project. 4. Fill out chapter 8,9 and the student's part in 10.	
	1. Work on chapter 5.3, create sequence diagrams. 2. Fill out section 3.1.1. 3. Start working on non-functional requirements. 4. Write his own student reflection in chapter 10.	Nawaf Al-Sowadi Abdulaziz Al-Kubaisi
Final Milestone: Finalizing the whole project.	Finalizing the project and submitting it.	Sultan Alemadi Nawaf Al-Sowadi Abdulaziz Al-Kubaisi

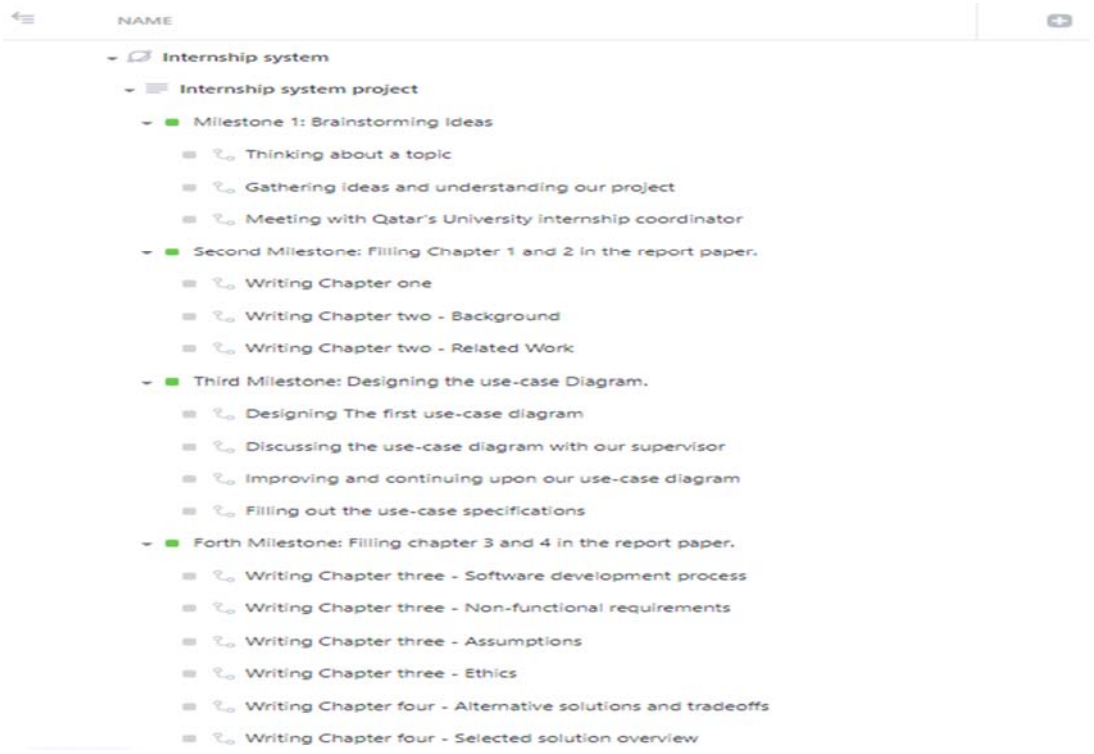


Figure 39. Project Timeline

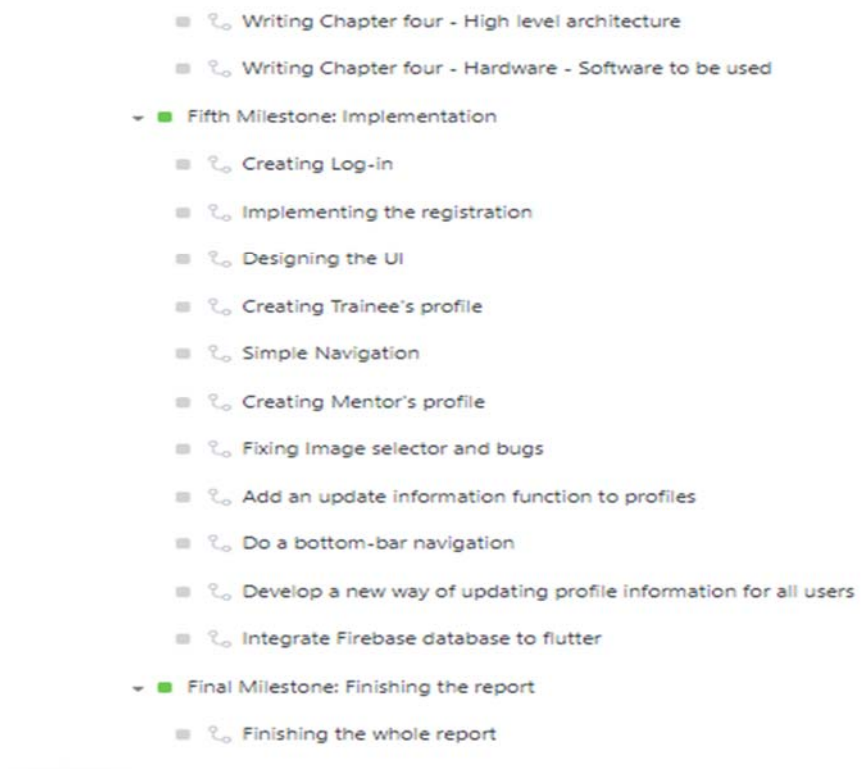


Figure 40. Project timeline (2)

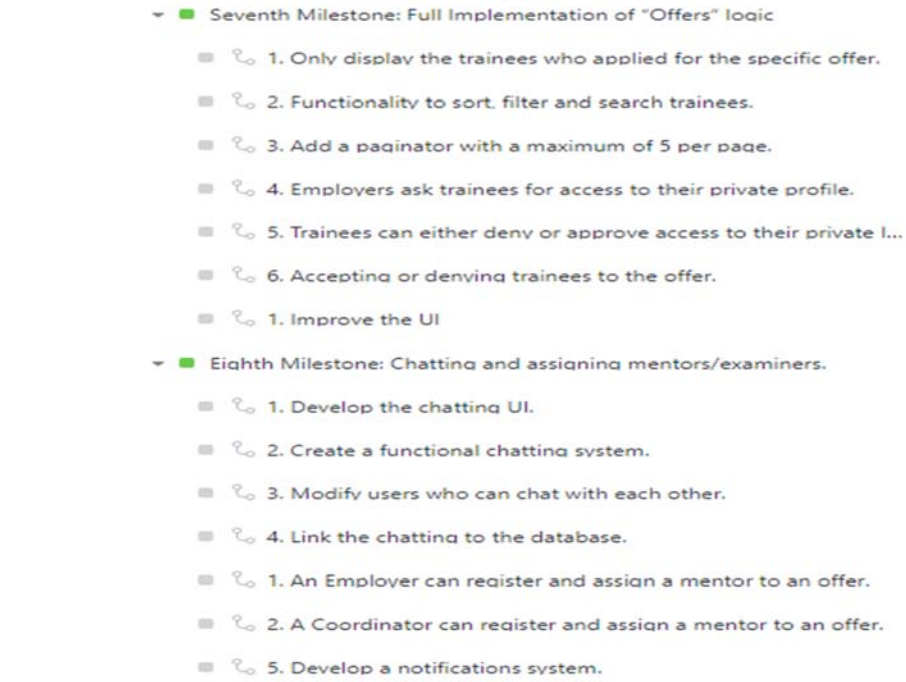


Figure 41. Project timeline (3)

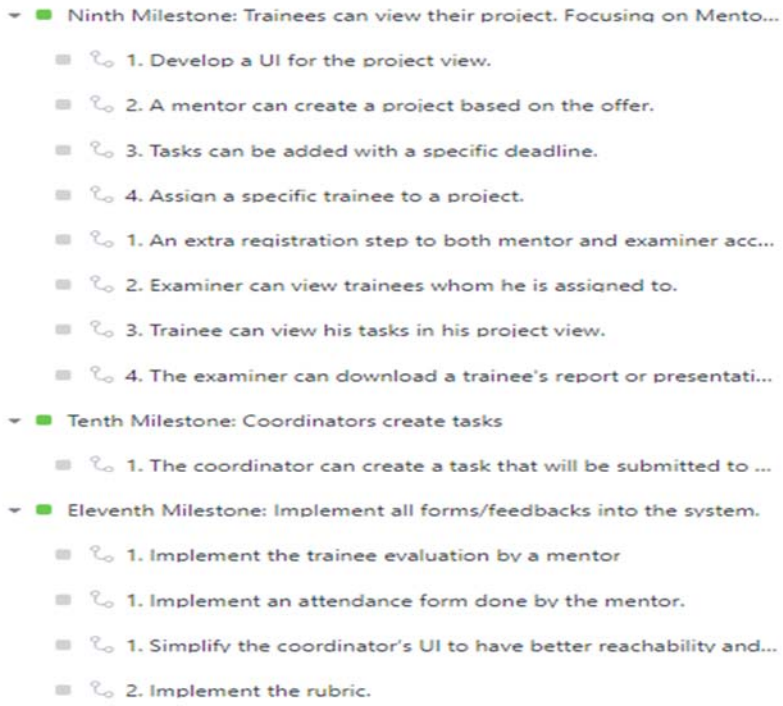


Figure 42. Project timeline (4)

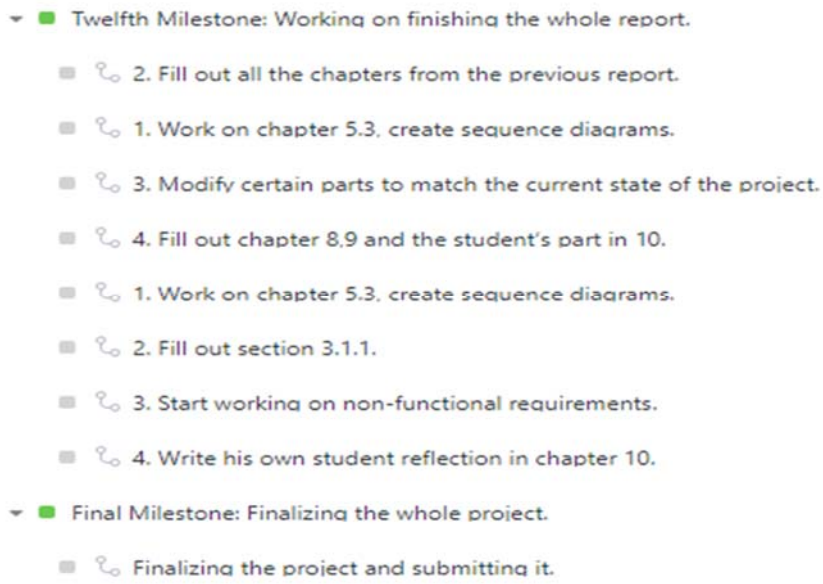


Figure 43. Project timeline (5)

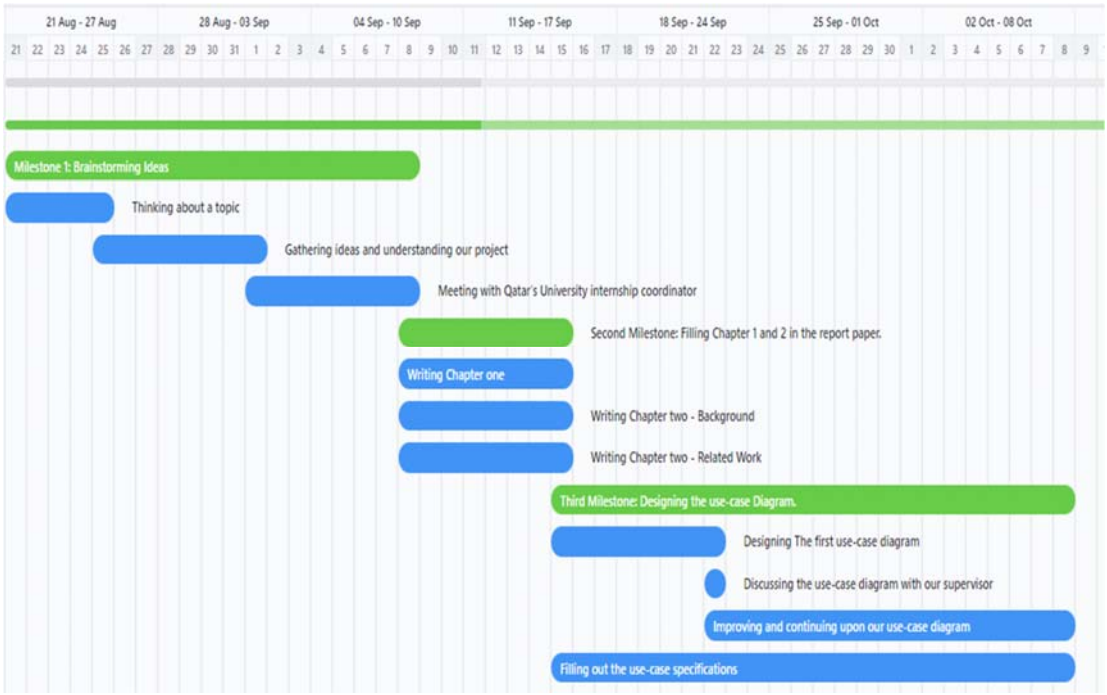


Figure 44. Project timeline (6)

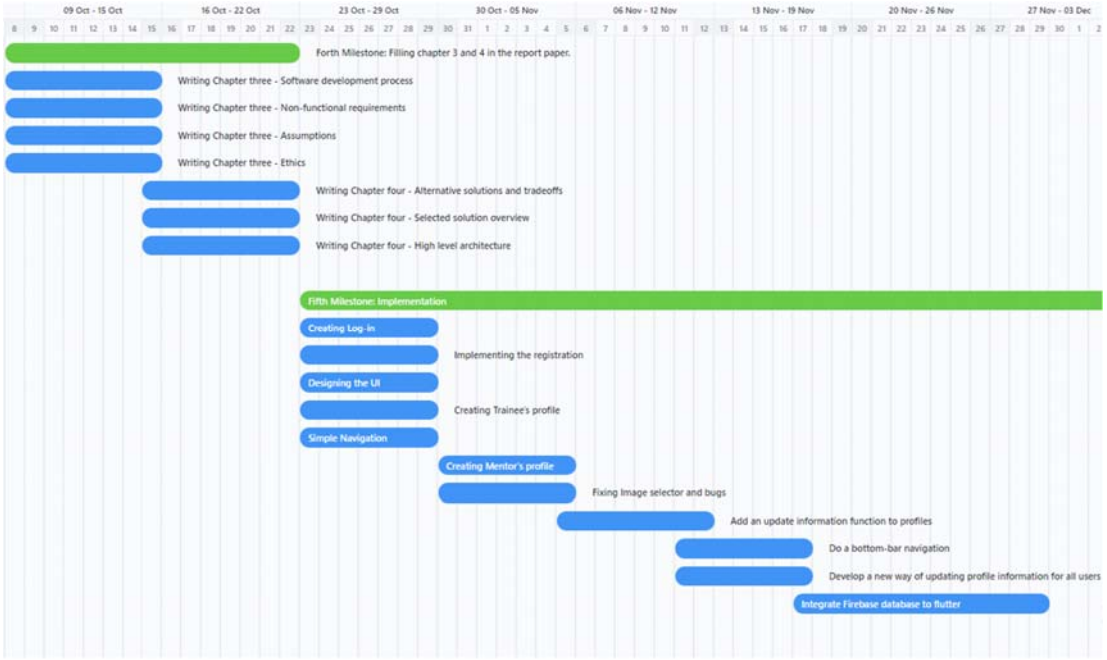


Figure 45. Project timeline (7)

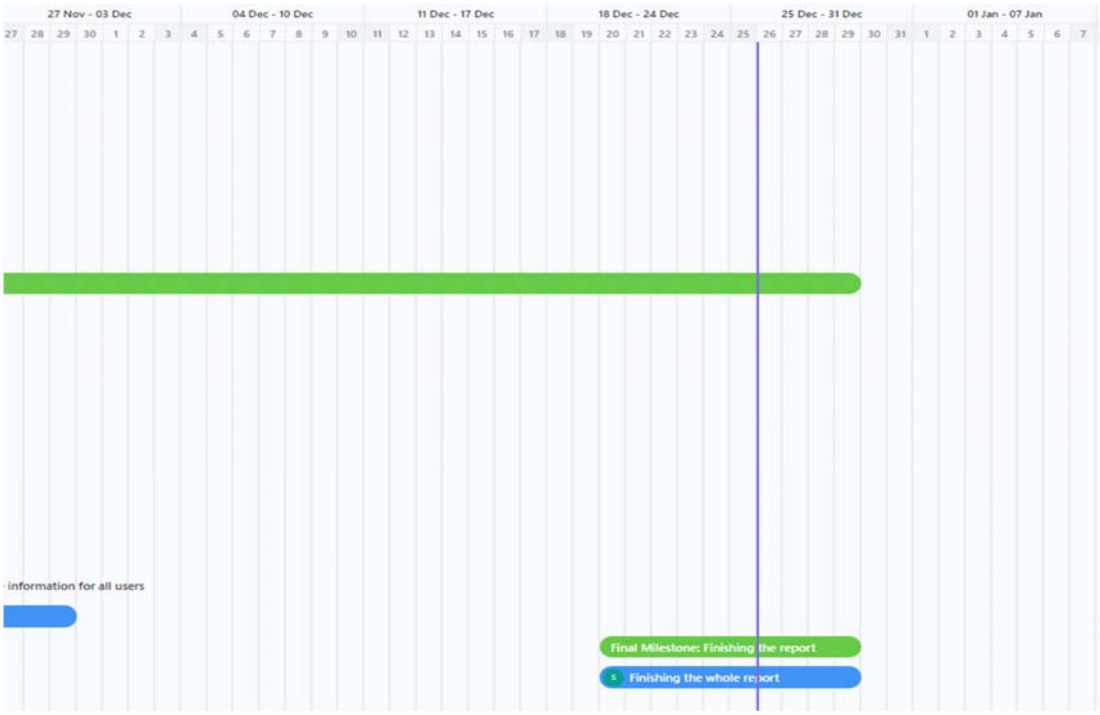


Figure 46. Project timeline (8)

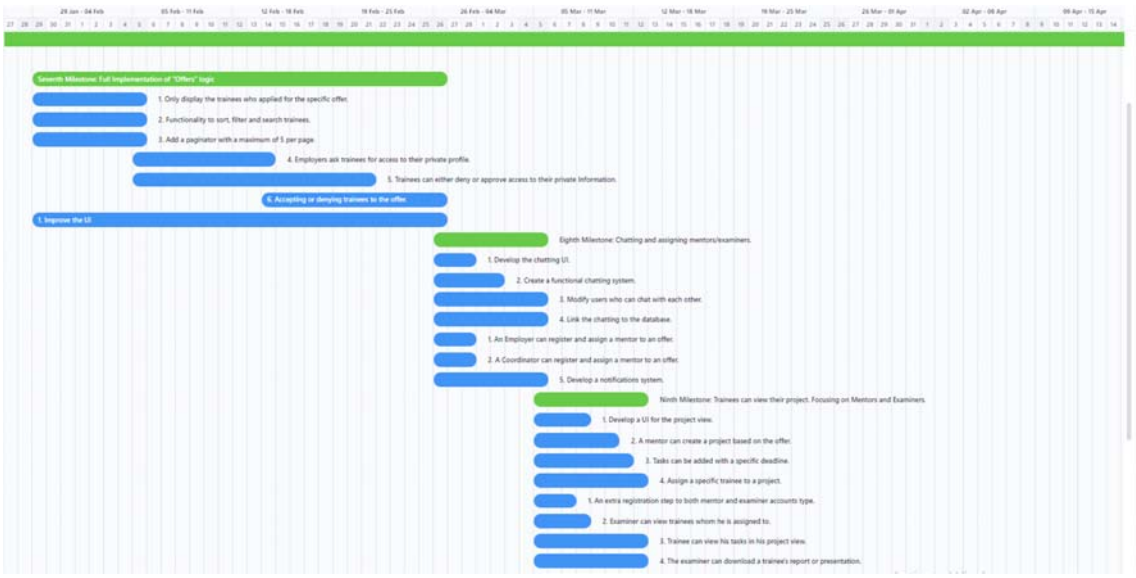


Figure 47. Project timeline (9)

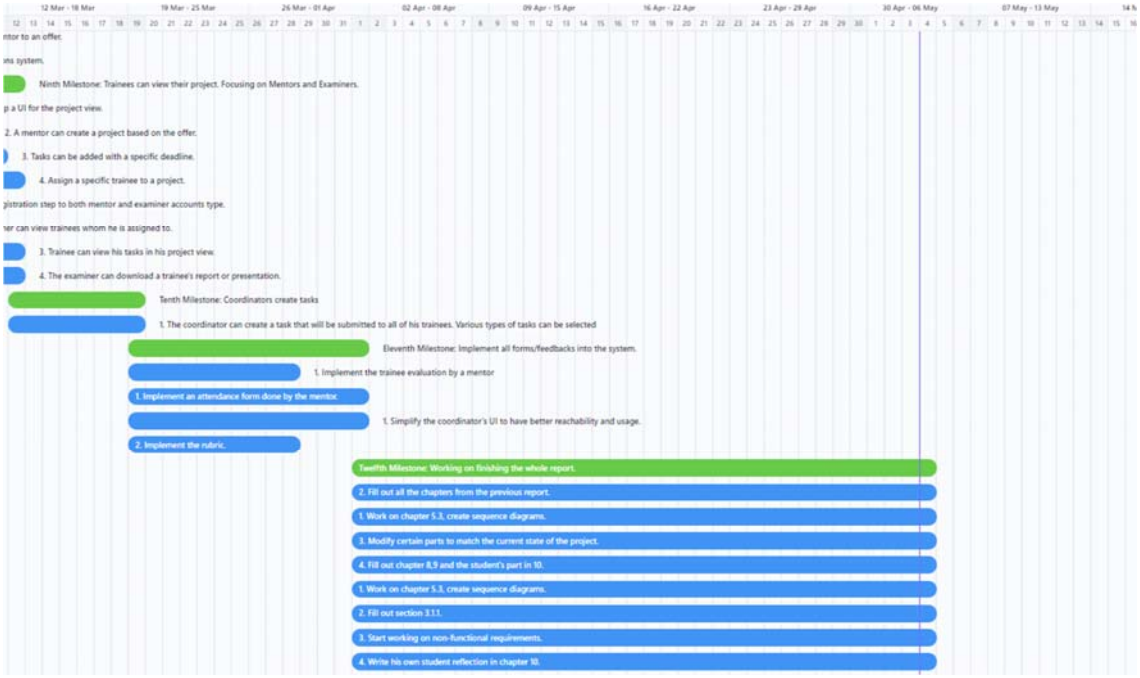


Figure 48. Project timeline (10)

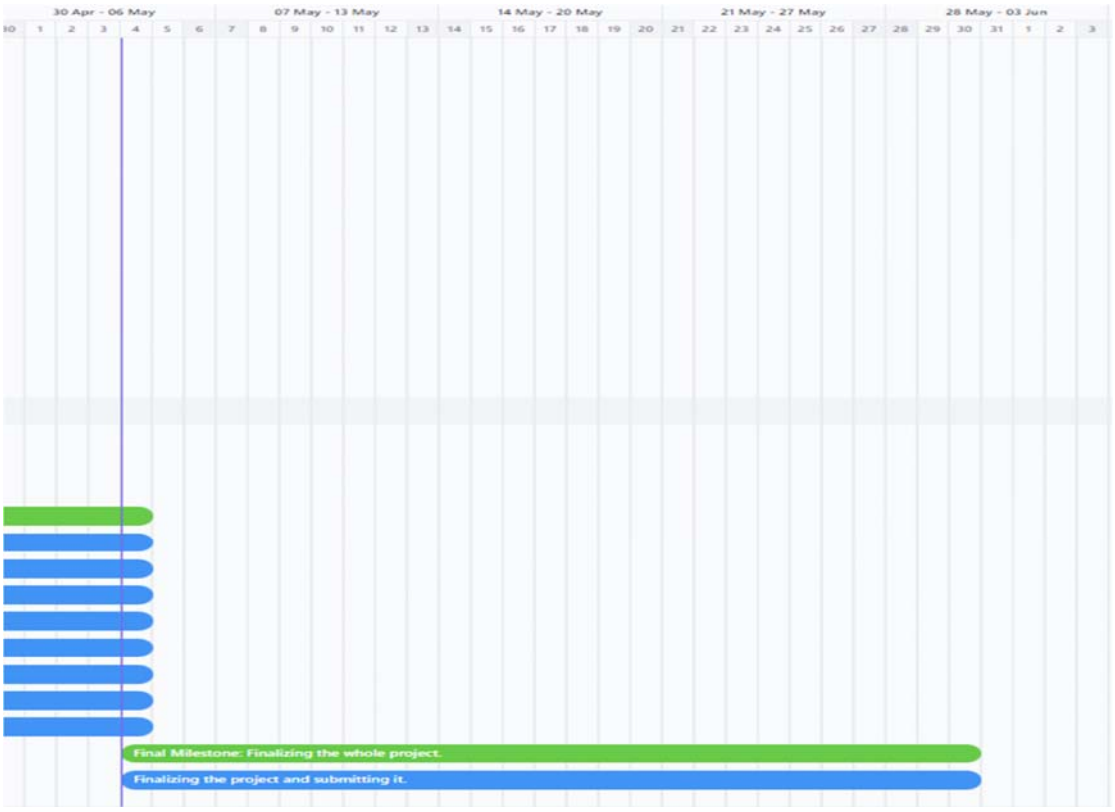


Figure 49. Project timeline (11)

4.3. Anticipated risks

Table 8 outlines the foreseeable challenges we might encounter in our project, including limited Dart language skills, meeting deadlines, securing cooperation from employers and coordinators, managing data storage, and ensuring wide accessibility of our project. For each of these challenges, the table provides our plans to lessen their impact, demonstrating our preparedness to navigate these potential issues, and maintain our project's successful progress.

Table 8. Anticipated Risks

Risk event	Approach to minimizing the effect on project success
Lack of experience in using Dart language	Every team member had to practice by doing tasks that covered up the most important basics of the language. And a lot of learning resources have been shared to gain knowledge.
Missing the deadline	We as a team had an unofficial deadline which was on Saturday of every week. Starting each week, we distribute the work among us and no matter what we must deliver our tasks and present it to each other before the deadline.
No cooperation from employers nor coordinators	We met with an internship coordinator to make sure about the importance and reach of our project. And they said that it is extremely valuable, and many employers and coordinators are willing to use it as soon as possible.
Storing data	Data is the pinnacle of our project and we had to find a suitable database that can work with Dart language. After researching all the available options, we have found that firebase DB is a perfect modern pick, and it is also supported by google.
Reachability	Dart/Flutter solved one of our main issues that is the lack of reception. With these tools, we can have our application run nearly everywhere with the same base code.

5. Solution Design

5.1. High-level design

5.1.1. Alternative solutions and tradeoffs

In our case, we considered numerous alternative solutions. We ultimately opted for Flutter to address the challenges encountered by trainees searching for internships, as it proved to be the most superior option among the available choices. Before finalizing our decision, we evaluated the following solutions:

4.1.1 Web application by JS, HTML, CSS

This one potential solution was particularly strong due to its compatibility with all platforms. Our team members were already familiar with this option, which would have allowed us to focus more on enhancing other aspects of the project. Furthermore, the solution enjoys widespread popularity,

offering an abundance of resources to draw upon. In addition, our university houses experts in this area who could assist us in case of any issues.

Why not this solution?

However, this solution was not chosen because it is limited to web applications, whereas another option allows for both mobile and web application development within a single codebase. This limitation prevents us from fully leveraging the benefits of mobile applications, such as faster operations, access to mobile features, and online/offline capabilities. Moreover, while some users prefer mobile applications, others favor web applications depending on their devices. By using a single-codebase solution, we can cater to both preferences.

4.1.2 Mobile application by Kotlin

The Kotlin mobile app development community has been growing rapidly recently. Kotlin has been designated as the second official language of Android systems by Google. Kotlin is a cross-platform, general-purpose programming language for the Java Virtual Machine [9]. It is a language that extensively borrows from Java, and one of its objectives was to match Java's fast compilation speed. Kotlin lists many of its advantages. For example, Kotlin programmers write fewer lines of code than other languages; it imposes no runtime overhead; it suits multi-platform development; and it has many other advantages. In addition, this language is close to Java, and Java is the most taught language at our university. Also, we have at our university experts in Kotlin, which could help us if we choose this solution.

Why not this solution?

However, Kotlin was not selected due to its limitation to the Android platform. This constraint not only restricts the solution to one type of application but also confines it to a single platform within that category. We aim to create an application that is universally accessible and beneficial to all users. By choosing Kotlin, we would have excluded a significant number of potential users, thereby increasing the likelihood of our application's failure. In contrast, our chosen solution, Flutter, effectively addresses these concerns.

5.1.2. Selected solution overview

Flutter

In today’s world, cross-platform mobile application development is a pressing priority. Developers are forced to either develop the same application numerous times for various operating systems or accept a low-quality solution that trades performance for portability [10]. After looking into all the available options, we landed on using Flutter’s framework because of the many benefits that align with our vision. Flutter is a cross-platform framework developed by Google. First released in May 2017, it continues its rapid growth and becomes more popular in the developers’ circles. Flutter will boost the population of our system because of its portability. With one code base we can launch our application and make it work on the two current and most used systems, which are Android and iOS. In addition, we can also implement our application on the web making it accessible to everyone. Flutter also provides great developer experience. When comparing it to web development for example, in flutter you only must learn a high-level language which is Dart instead of learning HTML, CSS and JavaScript. Moreover, flutter is extremely efficient compared to the other frameworks and that’s because of its extensive widget’s library which gives the developers access to

a pre-made functionality which will reduce the coding time. And one of the most prominent features is the hot-reload that is only exclusive to Flutter. This feature allows you to quickly view the effects of your changes without the need to recompile the code again, thus saving a tremendous amount of time and enables the developer to experiment more with ease. Finally, the personal gain of learning flutter will help us a lot because of its rise in popularity, the demand of capable developers will be high.

Firebase

We had multiple databases to choose from, but our pick fell on Firebase. To start, it is the best database that works with our framework, which is Flutter, it can be integrated easily and with no issues. Also, Firebase provides more than one option to store data: we have Firebase Storage, Firebase Fire store, and the Realtime Database. Each of these can be chosen based on the use case and the type of data to be stored thus making it a flexible database.

5.1.3. High level architecture

Of all the available architectures, we have found that the MVVM architecture is the most suitable and best architecture to go by in this project. Due to various advantages that we are seeking in software architecture. The MVVM architecture is basically a software design pattern that is structured to separate program logic and user interface [4]. It helps to break programs into modules, and this was the main selling point to us as a team. it becomes possible distribute developers in each area work on related items at the same time. Designers can work on the UI at the same time as developers are working on the code without needing to have them both work on the same files at the same time [5]. In addition, MVVM provides better testability because of its modularity. Meaning that we can test various parts of the whole system without having the hassle of facing trouble caused by unrelated components. Figure 50 displays the flow of how the MVVM works and details the usage of each component, while figure 51 showcases the classes that are supposed to be created under each component and overall, it gives a more insightful look to the architecture. #

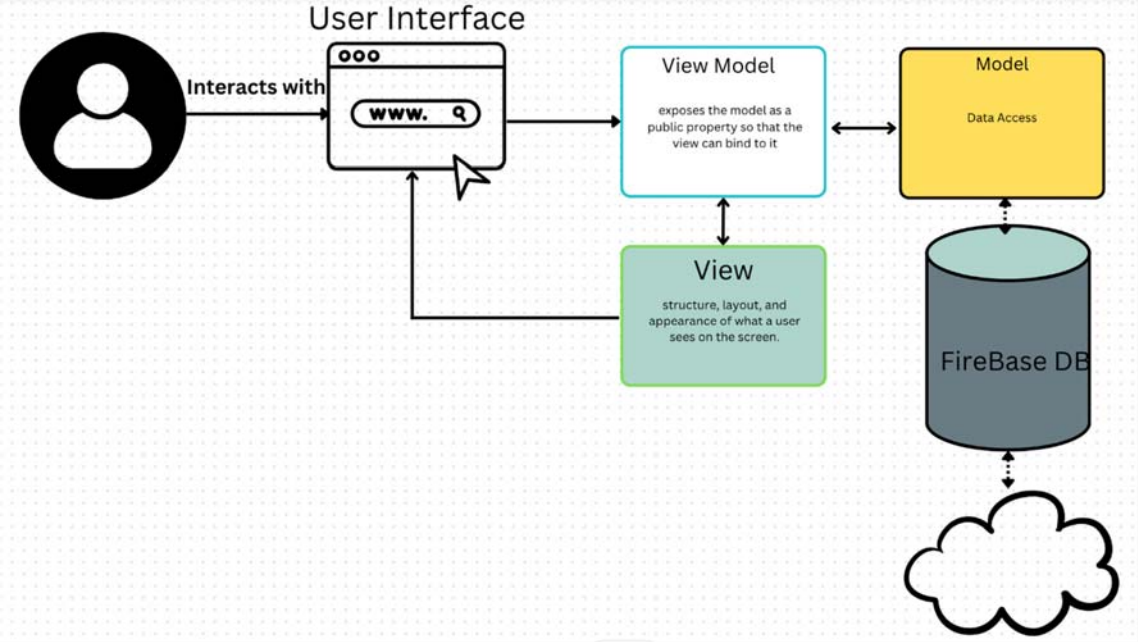


Figure 50. MVVM high-level architecture

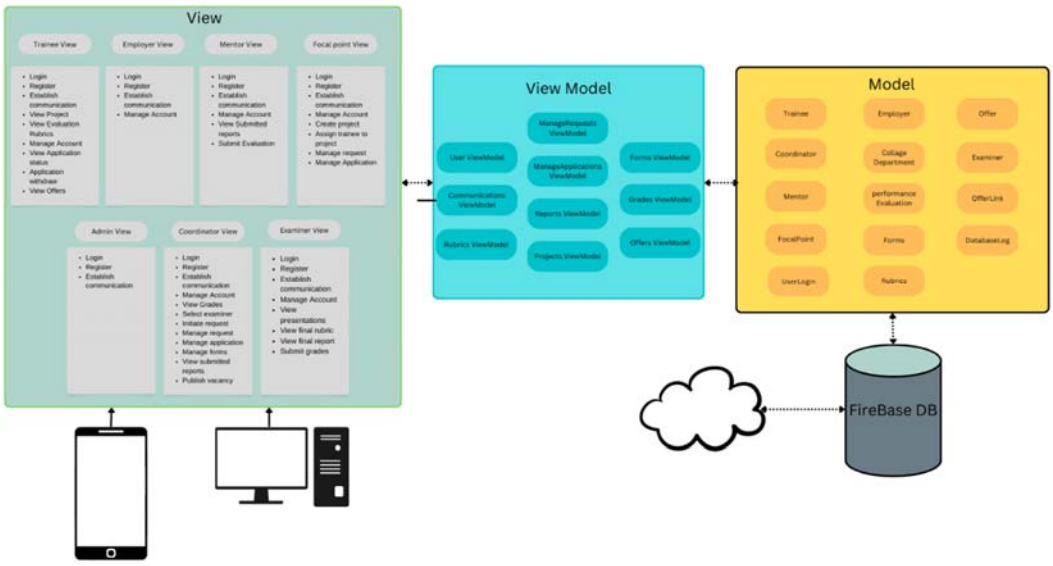


Figure 51. Detailed MVVM high-level architecture

Table 9. MVVM components

Component	Description
View	This layer represents what the user can view and interact with. As of now our system will be having 7 different views, each one is unique depending on the user type. Some similarities will be shared such as the main theme and logical design, but it might differ a little bit in functionality because each different user type will have access to some options that are not generally available. The view will be displaying the data from the View Model.
View Model	The View Model layer is the middle layer that connects the view and the model. It is responsible for exposing (converting) the data objects from the model in such a way they can be easily managed and presented. This layer also fetches the requested data from the model on behalf of the user's request and sends them to the View to be displayed. It is also where the algorithms and the whole functionality run on. Ten View Models is the estimated number to achieve simplicity in our code with each View Model having a one-to-one or one-to-many relationships with the models.
Model	The model layer is where the content and data are stored in. It is also connected to our database, which is **Firebase**. The model component responds to the View Model by giving out the requested data. In Addition, each type of data is identified with its correct form whether it was an Integer or a string.

5.2. Structural model

The class diagram of our project is a crucial piece of our software documentation, providing a clear overview of the system's structure by illustrating its classes, attributes, functions, and relationships. A deep dive and analysis explaining our position will be discussed and justified in this section. We used plantUML extension to extract the class diagram and modified it so it can be readable. Our system has reached over 80 classes and 150,000 lines of code so some heavy editing happened to the class diagram to provide a simpler look.

The following class diagram we present embodies the 'Model' component in our application's MVVM (Model-View-View Model) architecture. It mostly consists of user models, forms, notifications, and other elements, serving as the core data and business logic layer of our software system. Each class starts with the datatypes that we used. The "Map" method in each class diagram is what translates the strings and objects to the firebase.

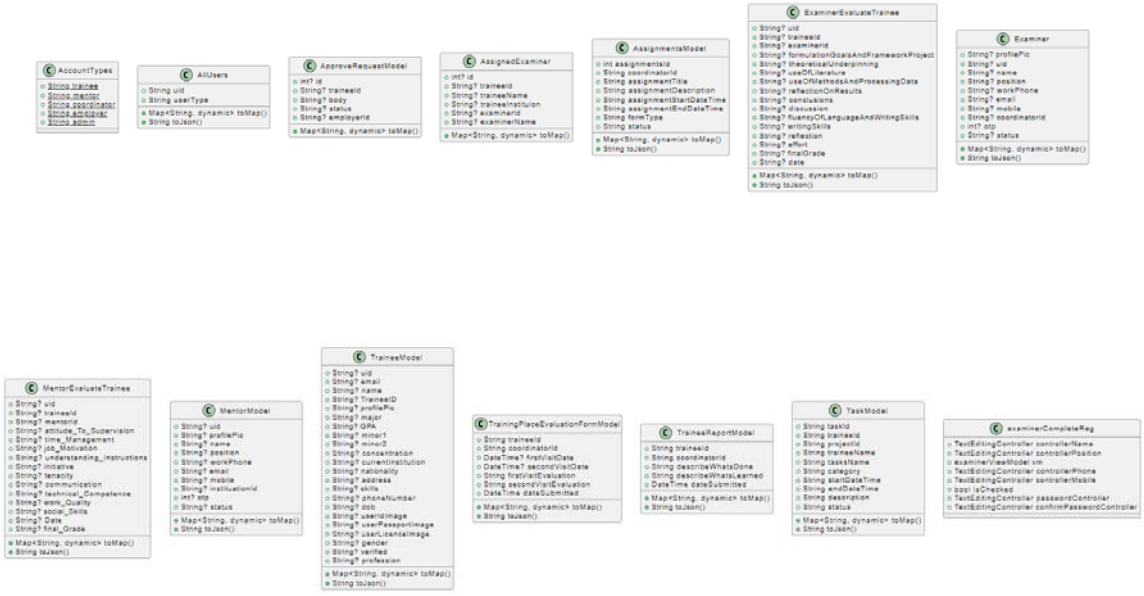


Figure 52. Class Diagram (1)

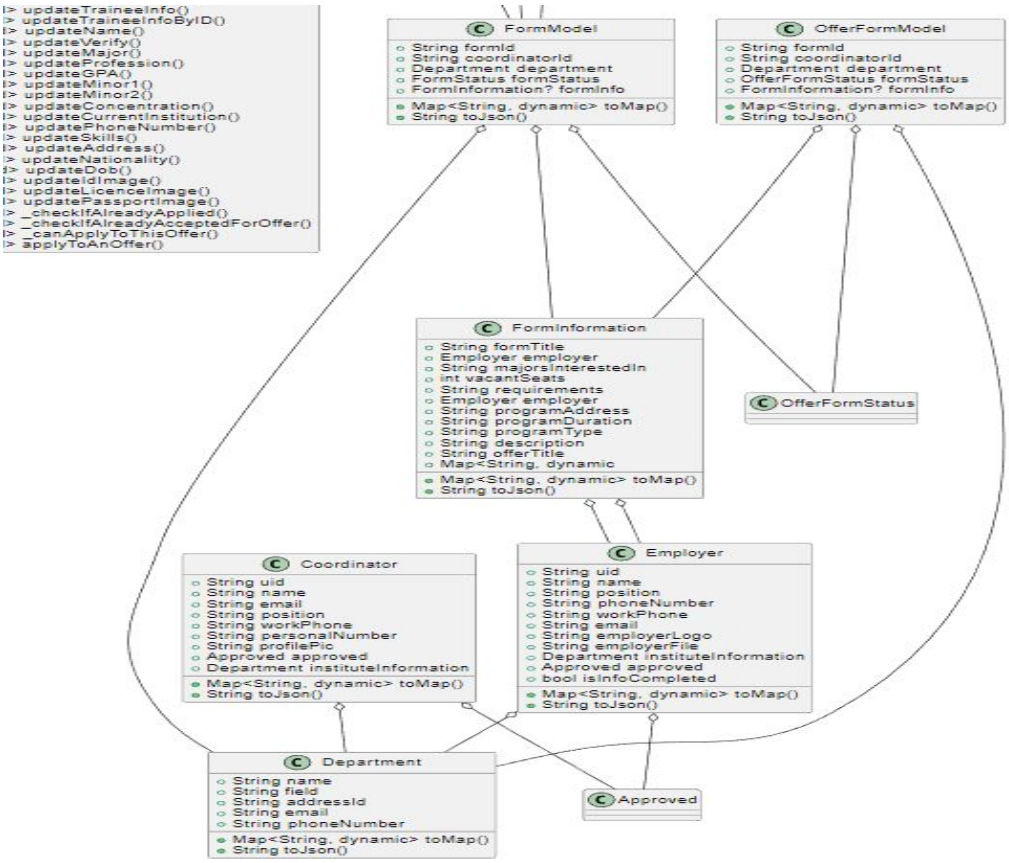


Figure 53. Class Diagram (2)

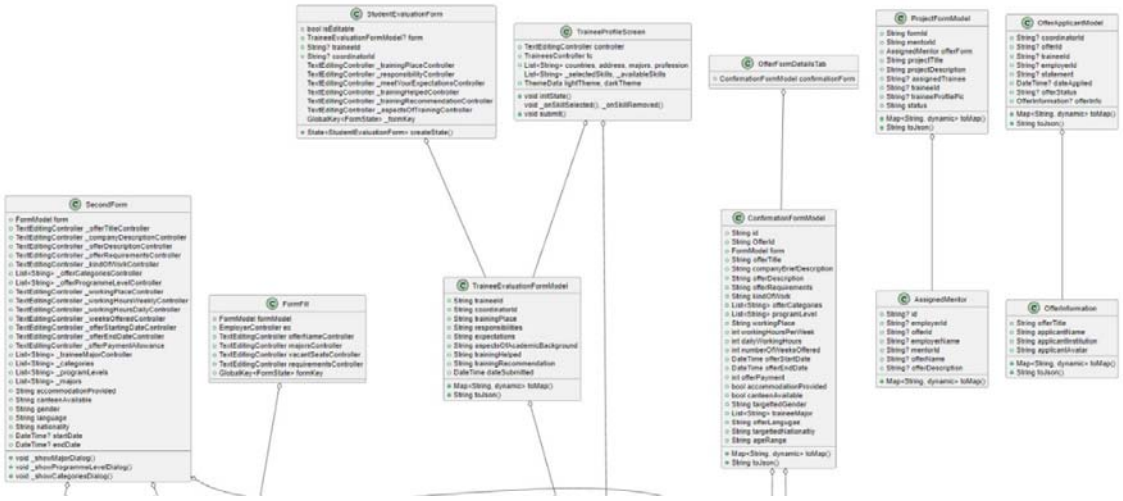


Figure 54. Class Diagram (3)

These two figures include the whole process of our “Offer” logic and the trainee evaluation submission. The Form model and the confirmation model are the two keys classes that are responsible for creating the final offer.

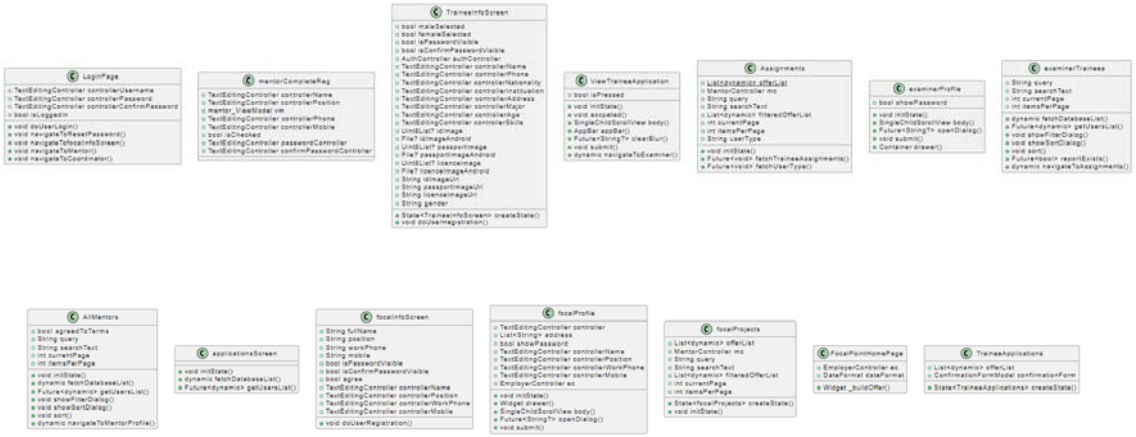


Figure 55. Class Diagram (4)

The class diagrams here represent mostly the examiner parts. Assignments is the class that displays trainees reports and presentations while giving him the ability to grade them.

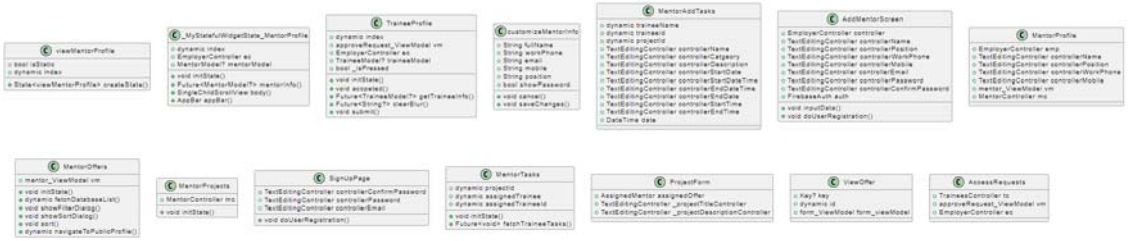


Figure 56. Class Diagram (5)

Mentors can add tasks that will display in trainees “MyProject” screen that reads both coordinator’s assignments and mentor’s tasks as we can clearly see in the class diagram, it fetches two times to retrieve both of them.

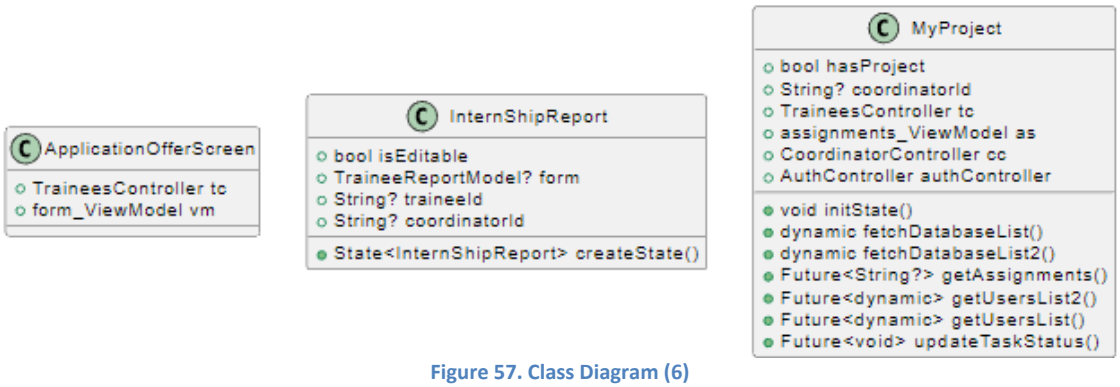


Figure 57. Class Diagram (6)

Figure 58. Class Diagram (7)

The Authentication class stores all the important methods regarding registration and login. Its one of the most crucial parts in our project, helping in securing the system and bridging the gap between it and the database.

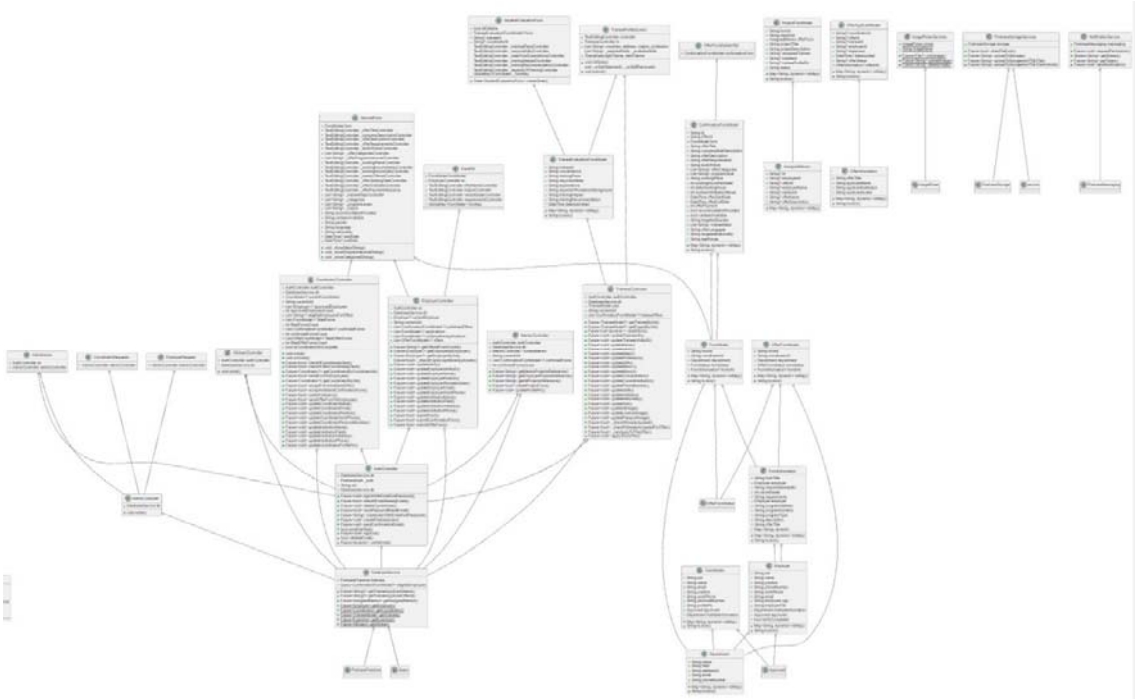


Figure 59. Class Diagram (8)

5.3. Behavioral model

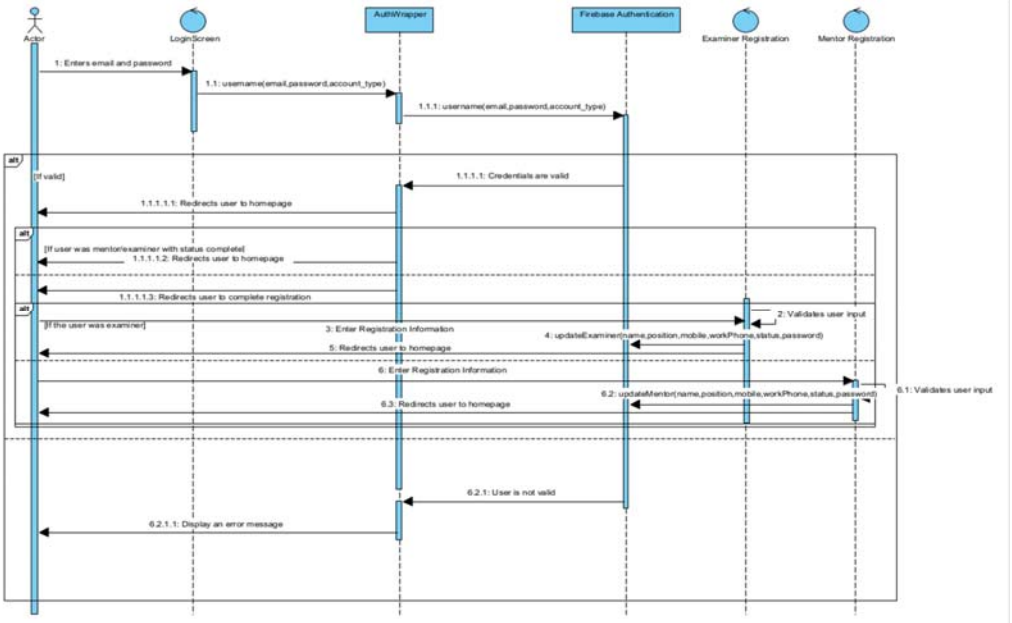


Figure 60. Login sequence diagram

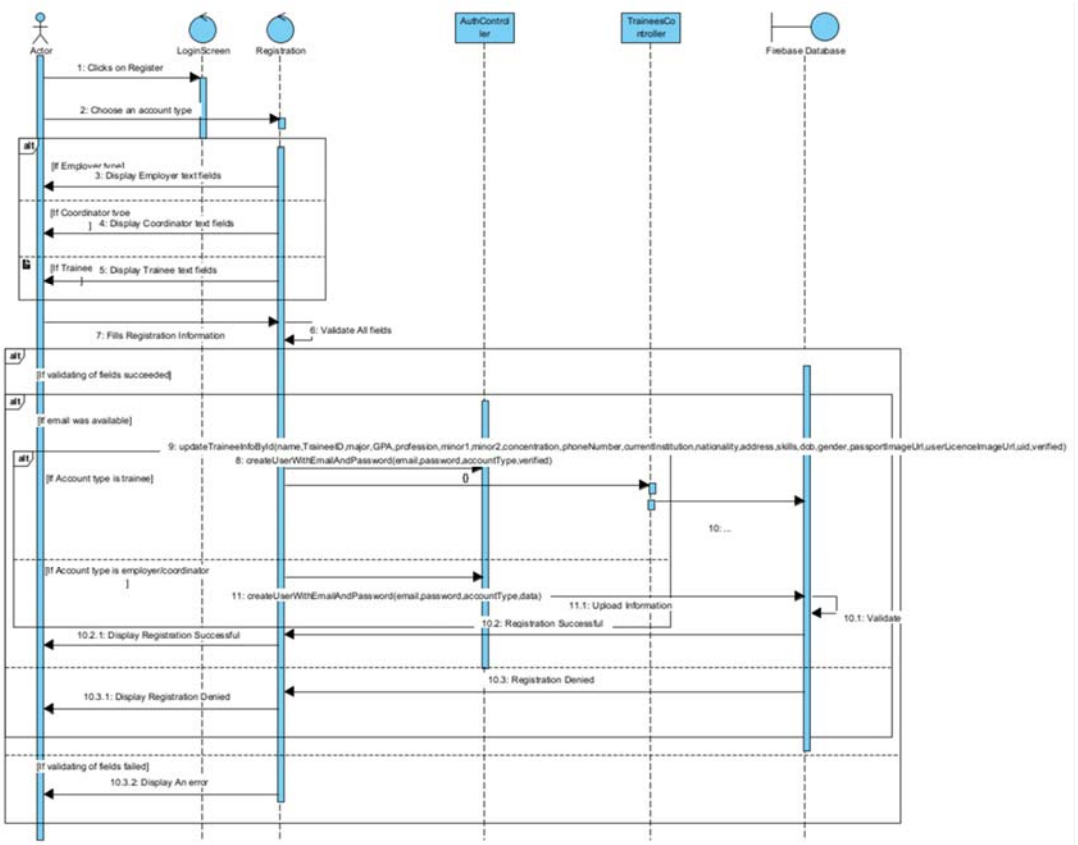


Figure 61. Registration sequence diagram

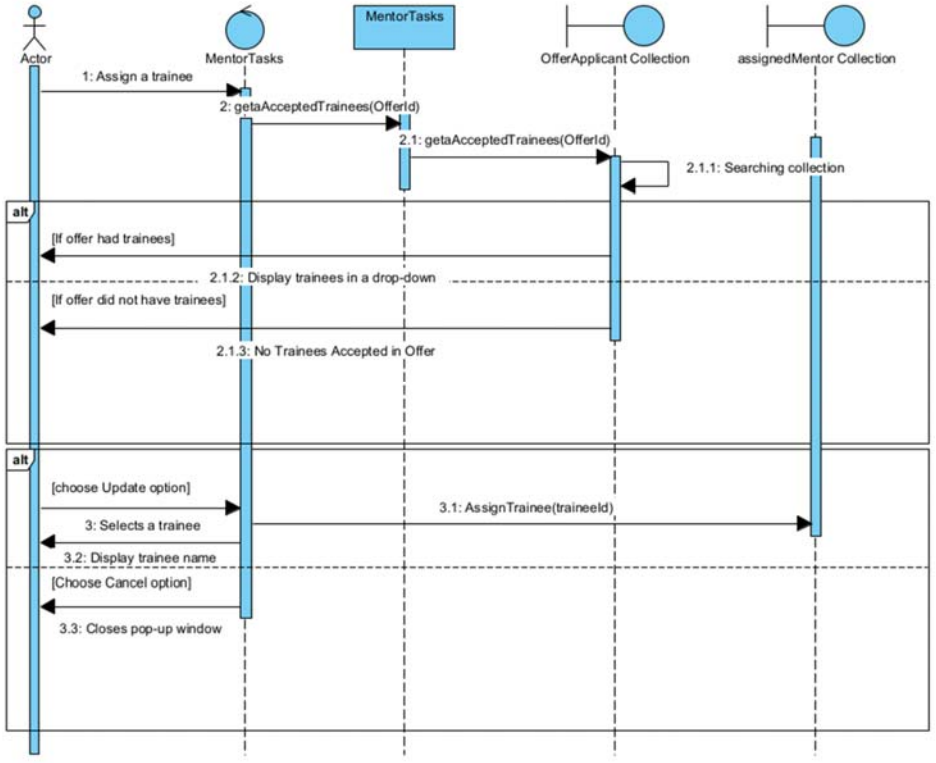


Figure 62. Assign a trainee sequence diagram.

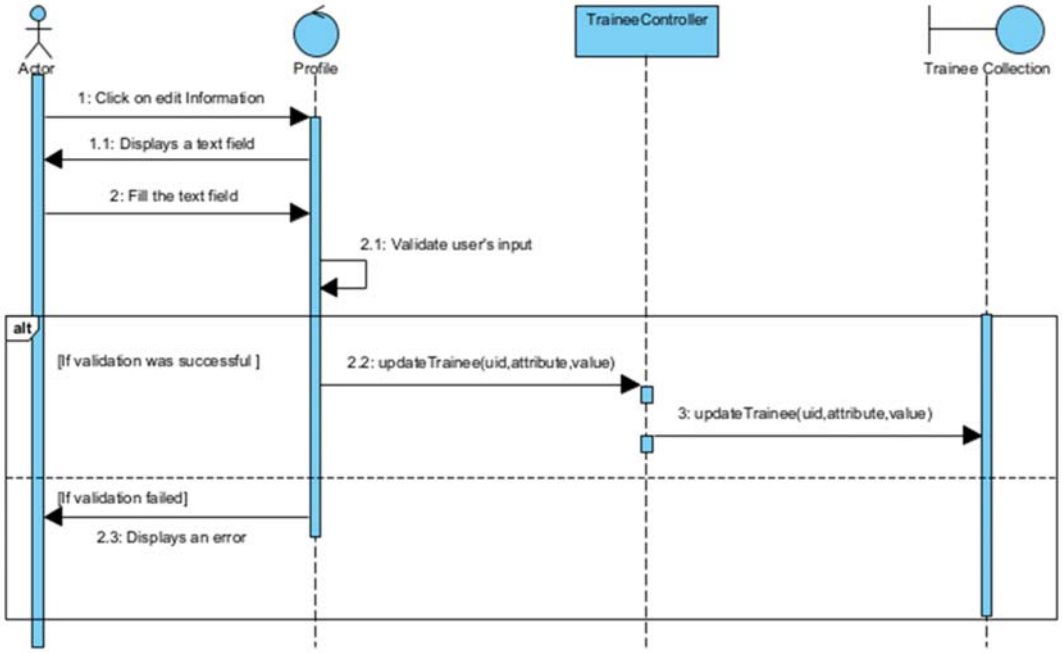


Figure 63. Customize Private Profile Sequence Diagram

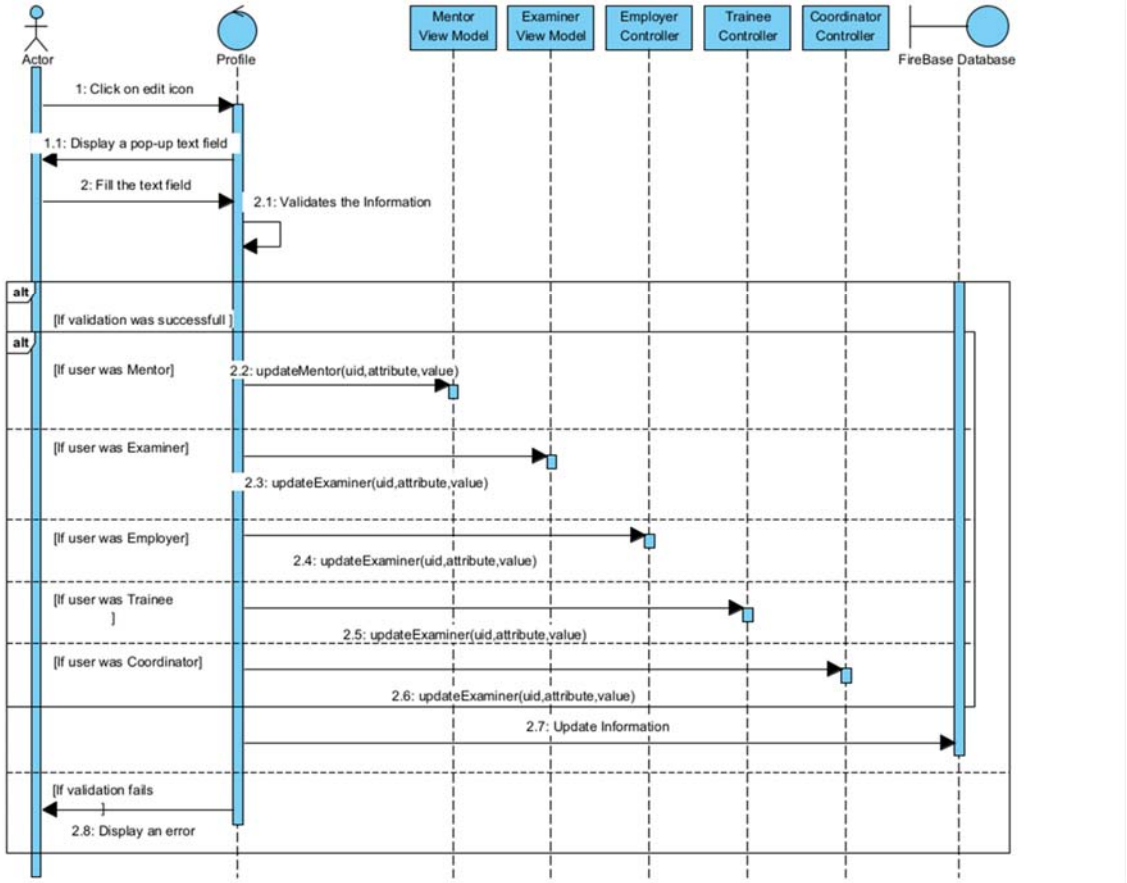


Figure 64. Customized Public Profile Sequence Diagram

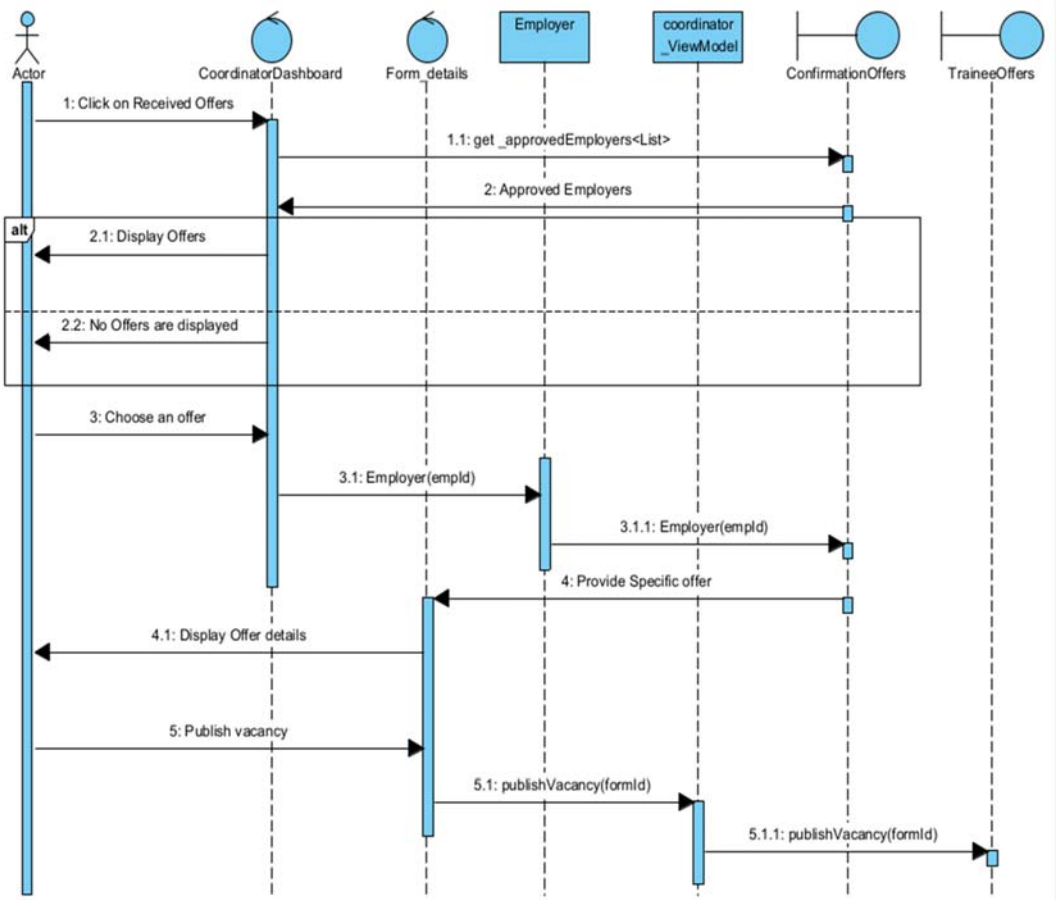


Figure 65. Publish Vacancy Sequence Diagram

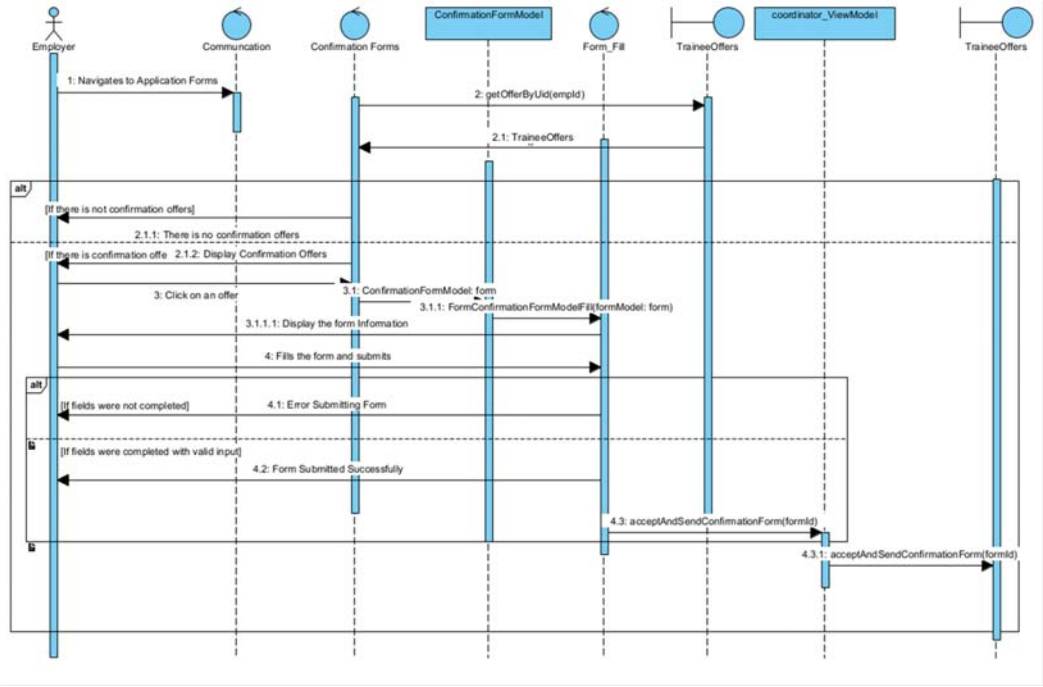


Figure 66. Manage Request Sequence Diagram

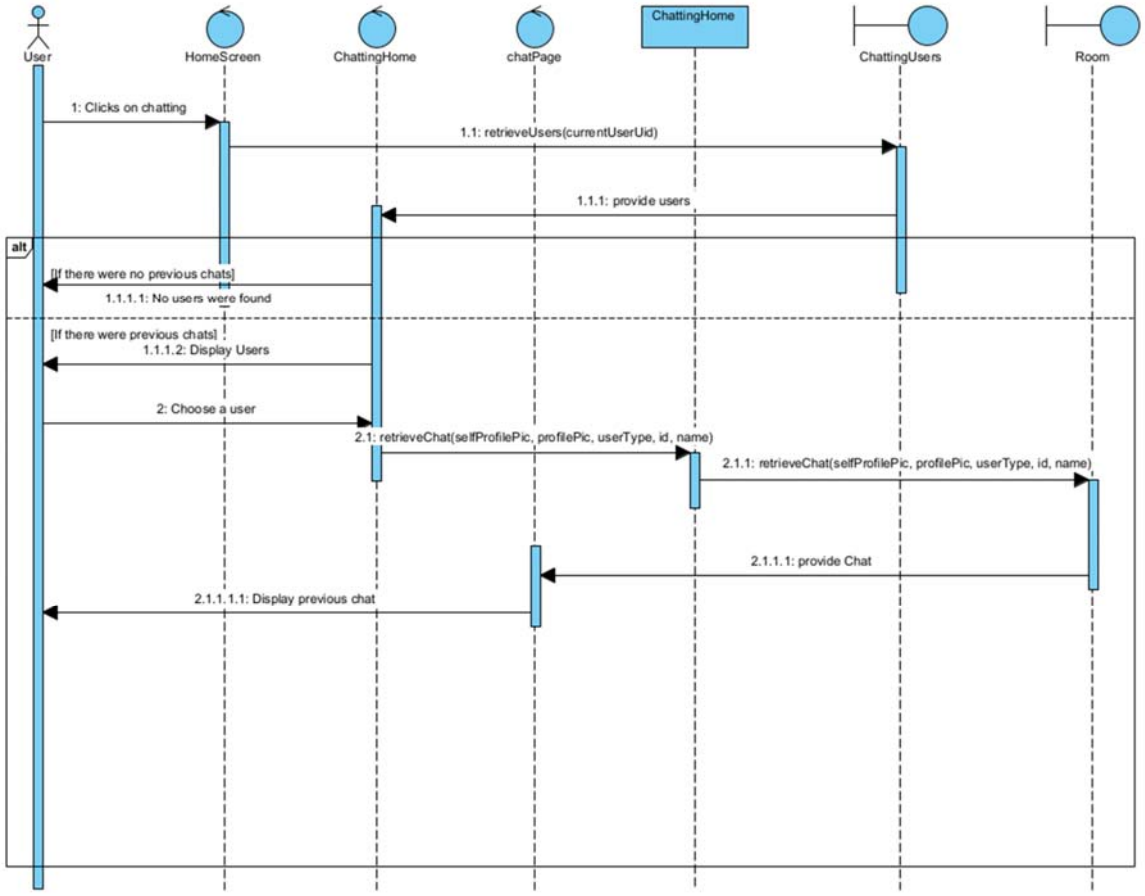


Figure 67. Establish Communication Sequence Diagram

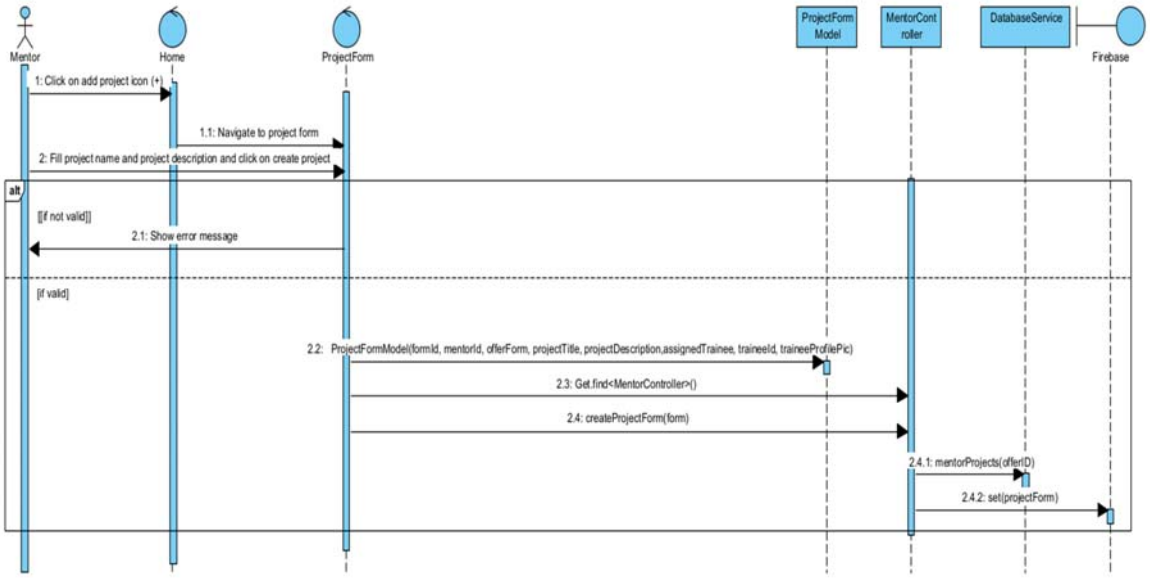


Figure 68. Create Project

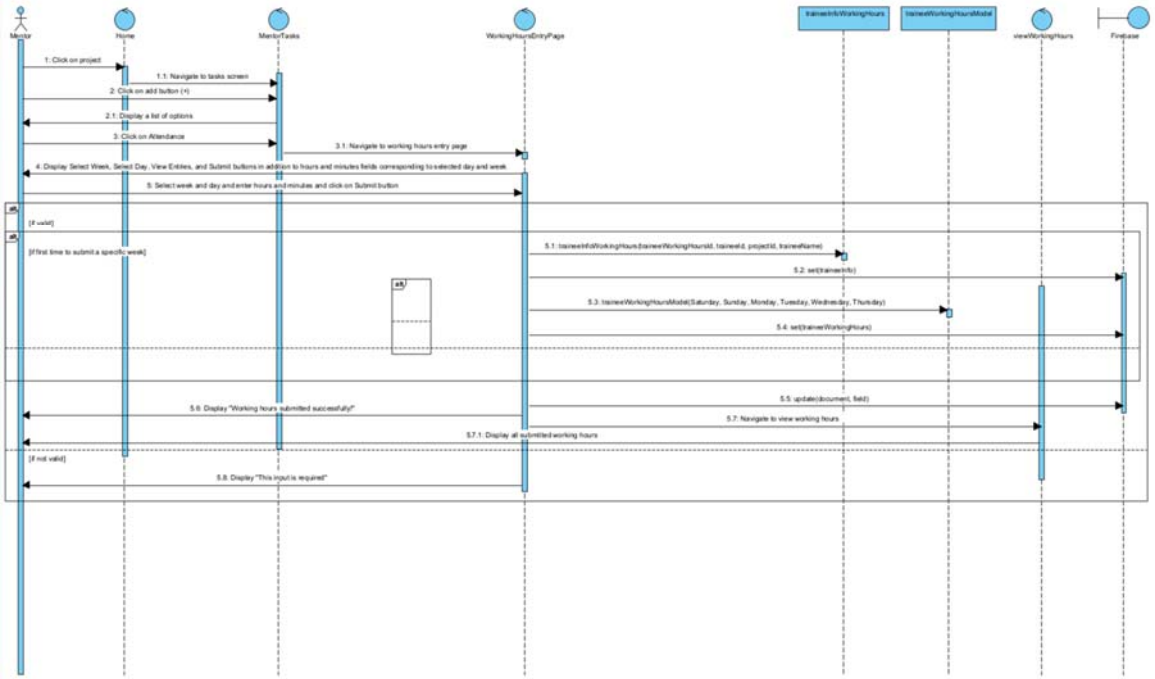


Figure 69. Trainee Working Hours

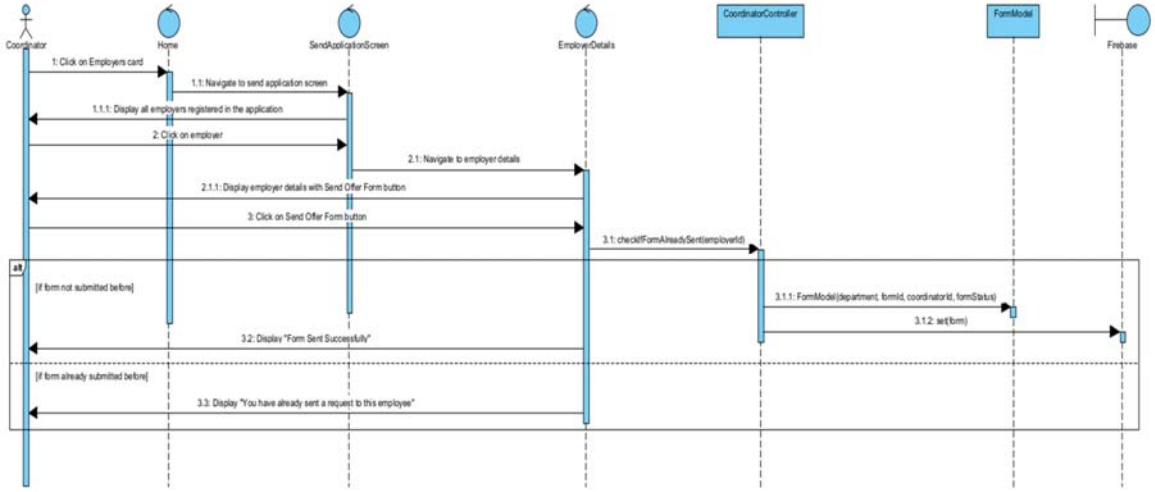


Figure 70. Initiate Request

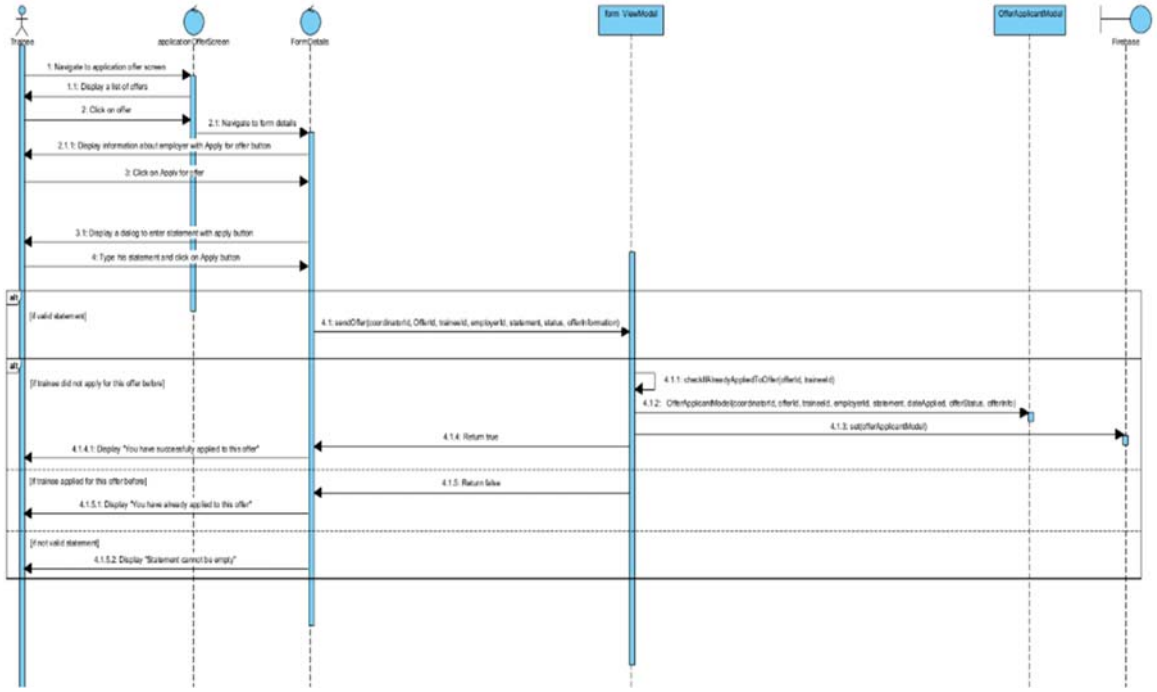


Figure 71. Apply for Offer

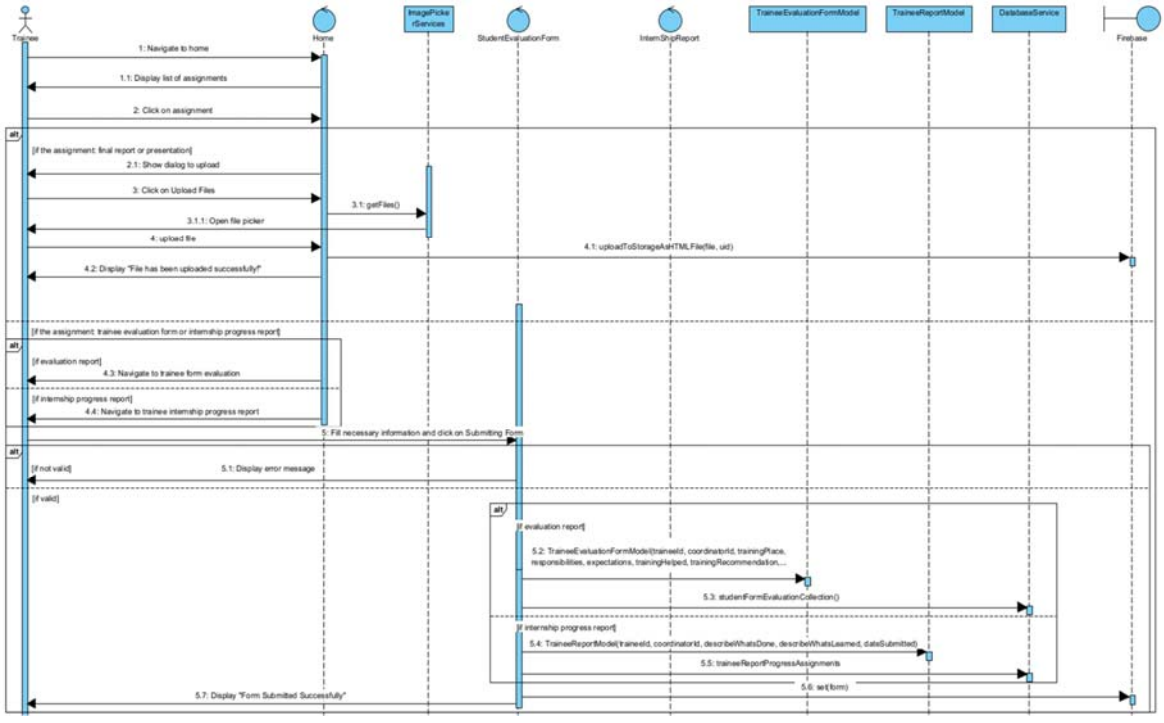


Figure 72. Submit Trainee Reports

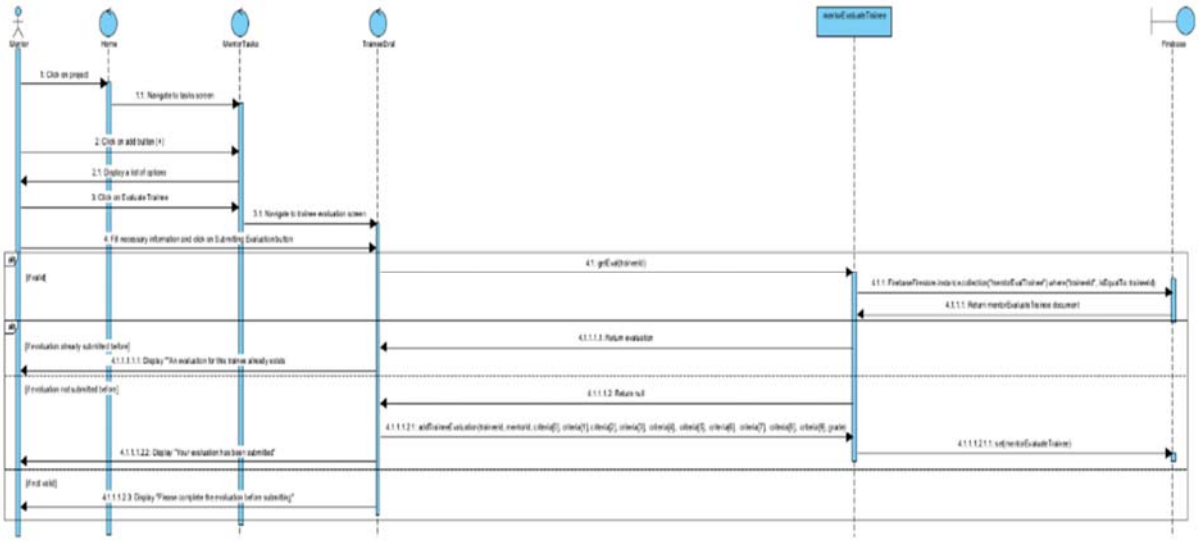


Figure 73. Mentor Evaluate Trainee

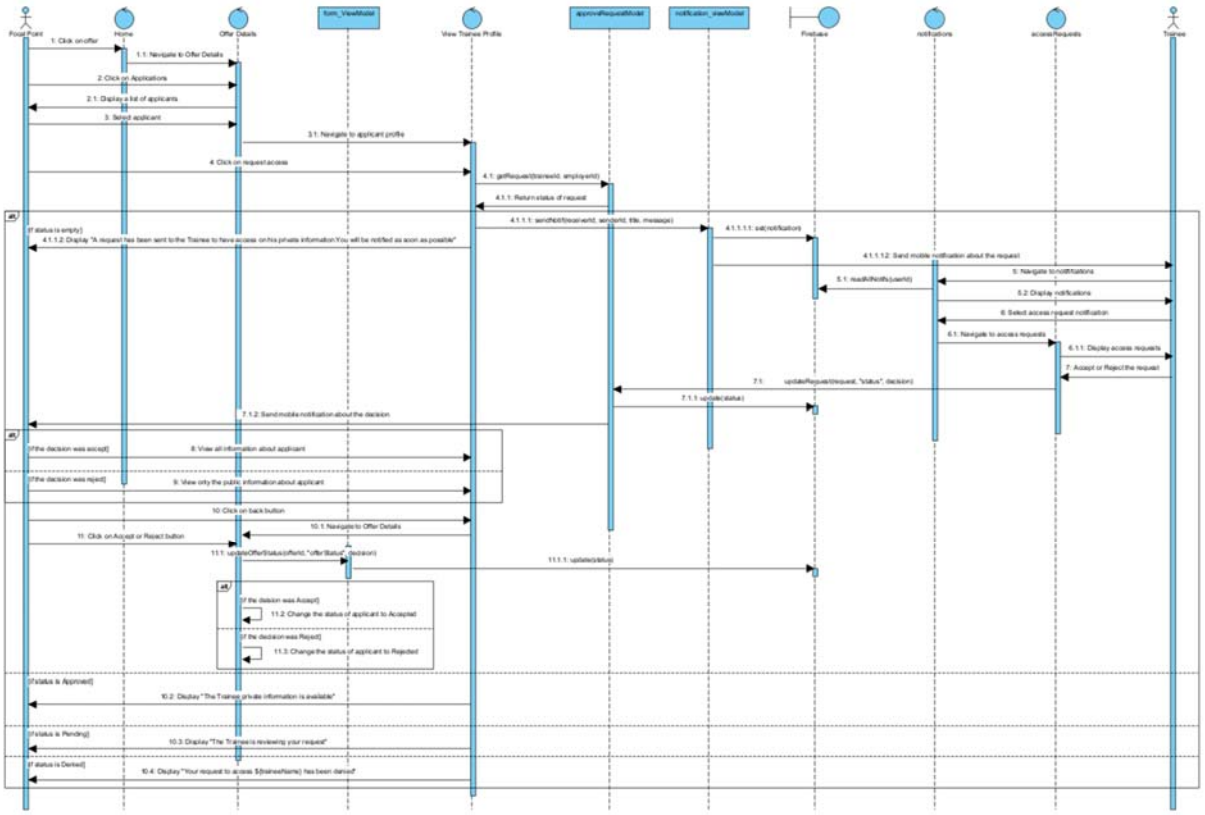


Figure 74. Manage Application

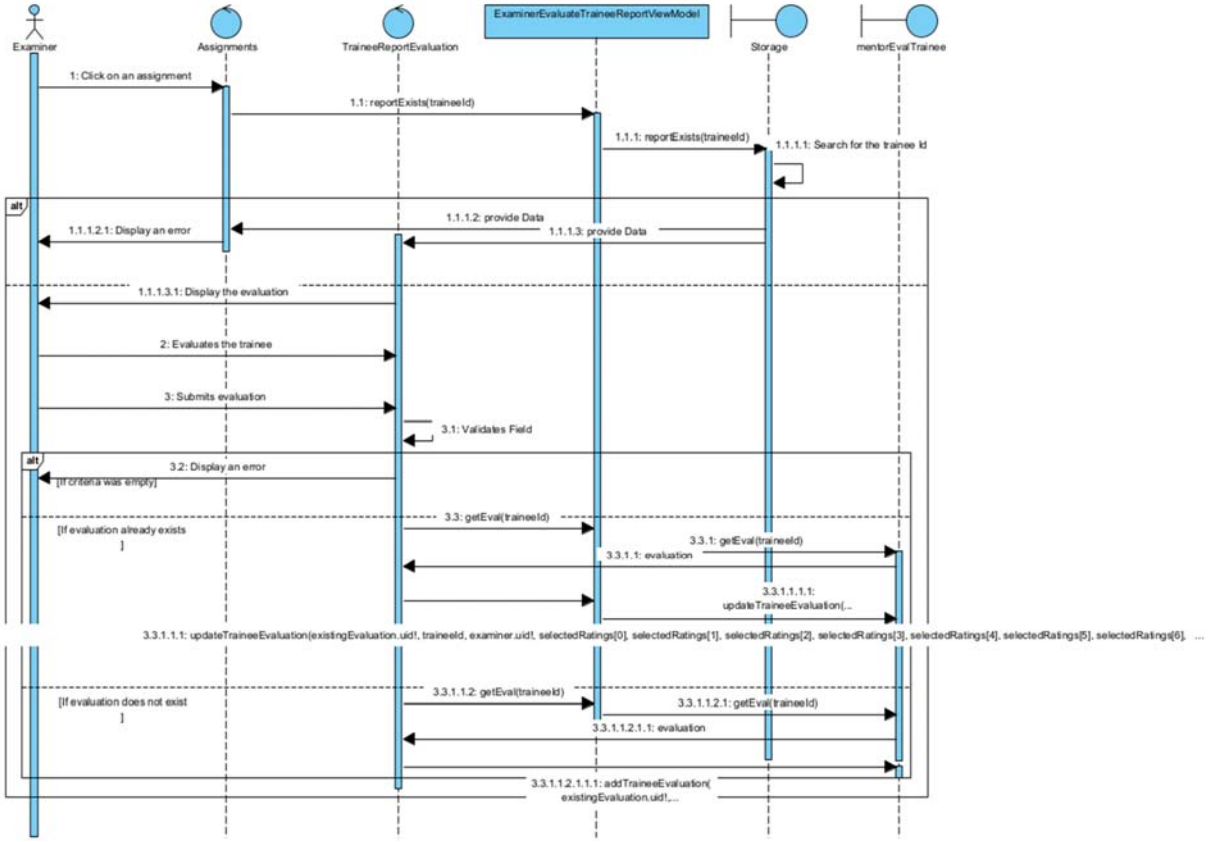


Figure 75. Submit Trainee Grades

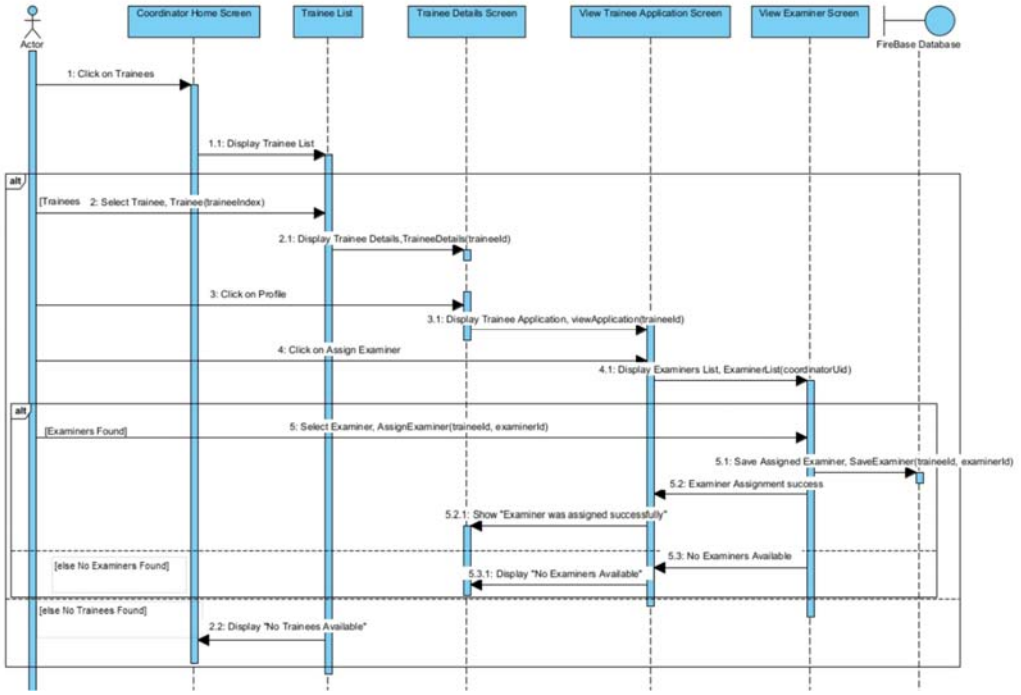


Figure 76. Assign Examiner

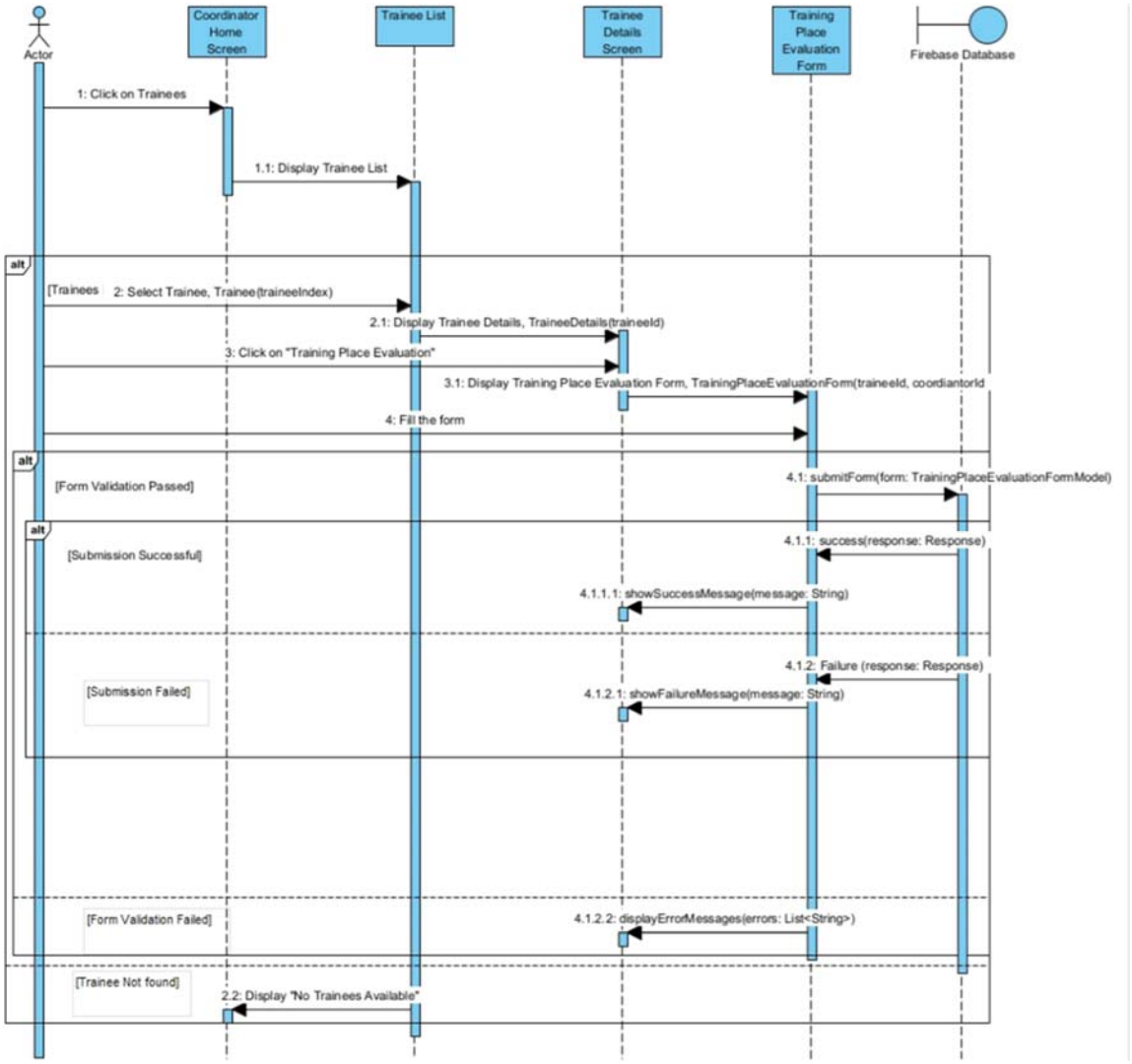


Figure 77. Training Place Evaluation

5.4. Database design

Our database diagram offers a comprehensive visualization of the structure of our internship system. Given its expansive nature, in figure 78 we've presented the complete diagram first, followed by figure 79 and figure 89 that are detailed snippets to enhance clarity. This diagram encapsulates all vital entities and information needed for our system. It includes the attributes of our key users - the trainees, coordinators, employers, mentors, and examiners - along with other integral components like chatting, forms, evaluations, and offers. The diagram is fundamental in illustrating how data is organized and interacts within our system. The relationships in our database diagram showcase how complex and interconnected our system is. They are thoughtfully established and carefully designed to reflect the dynamics among different entities. These relationships not only enhance the system's functionality but also contribute to its overall efficiency.

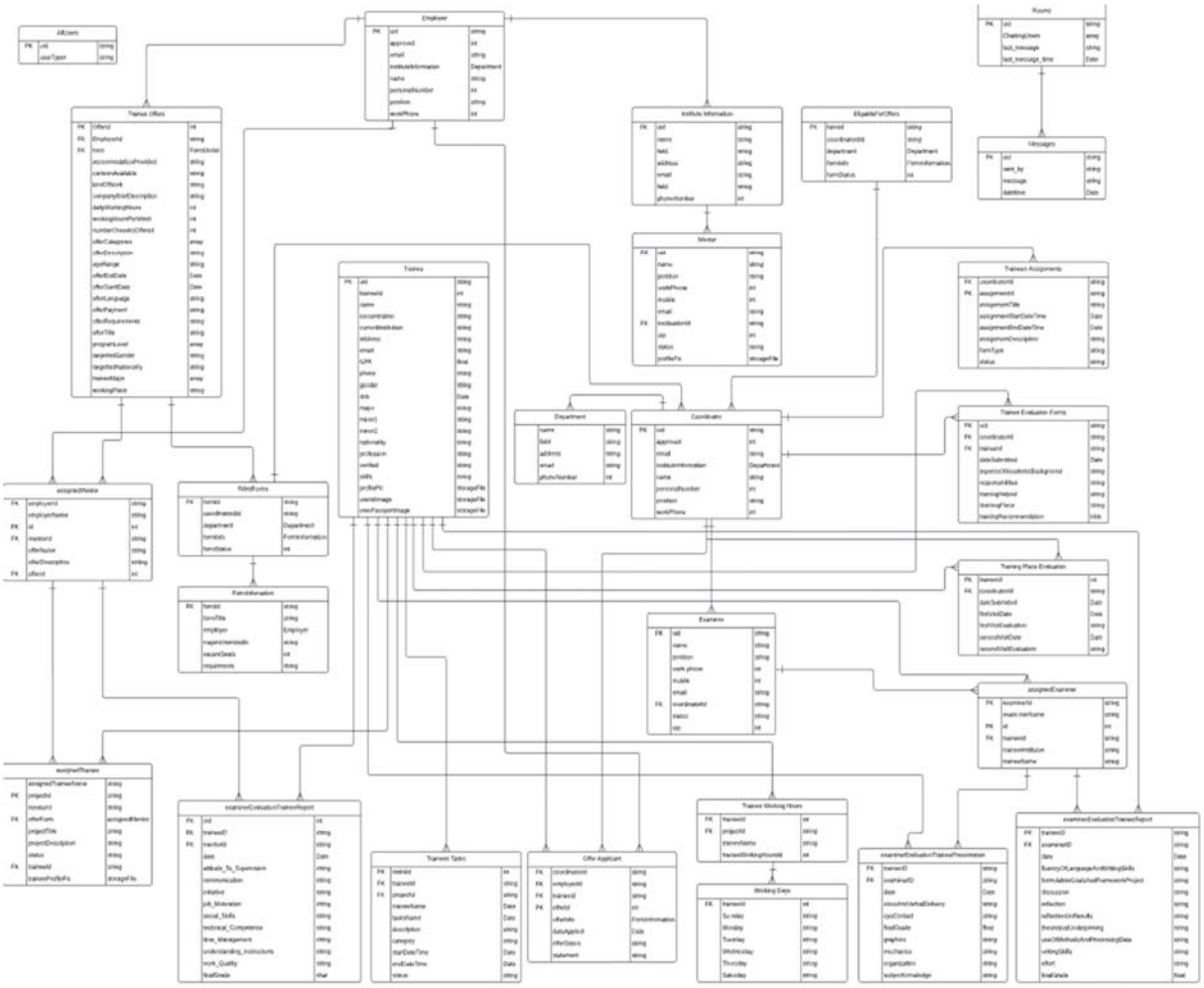


Figure 78. Database Diagram Full

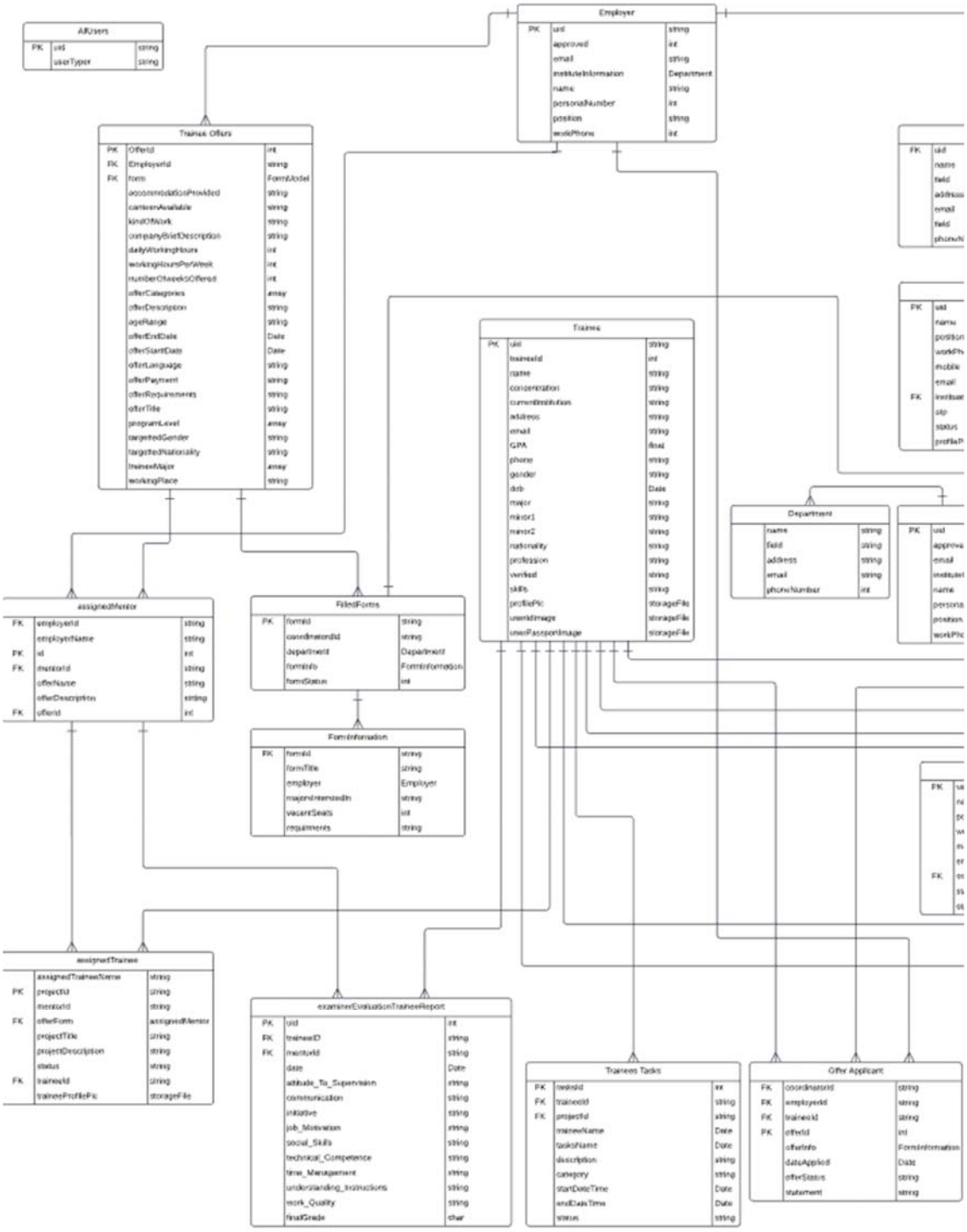


Figure 79. Database Diagram Snippet (1)

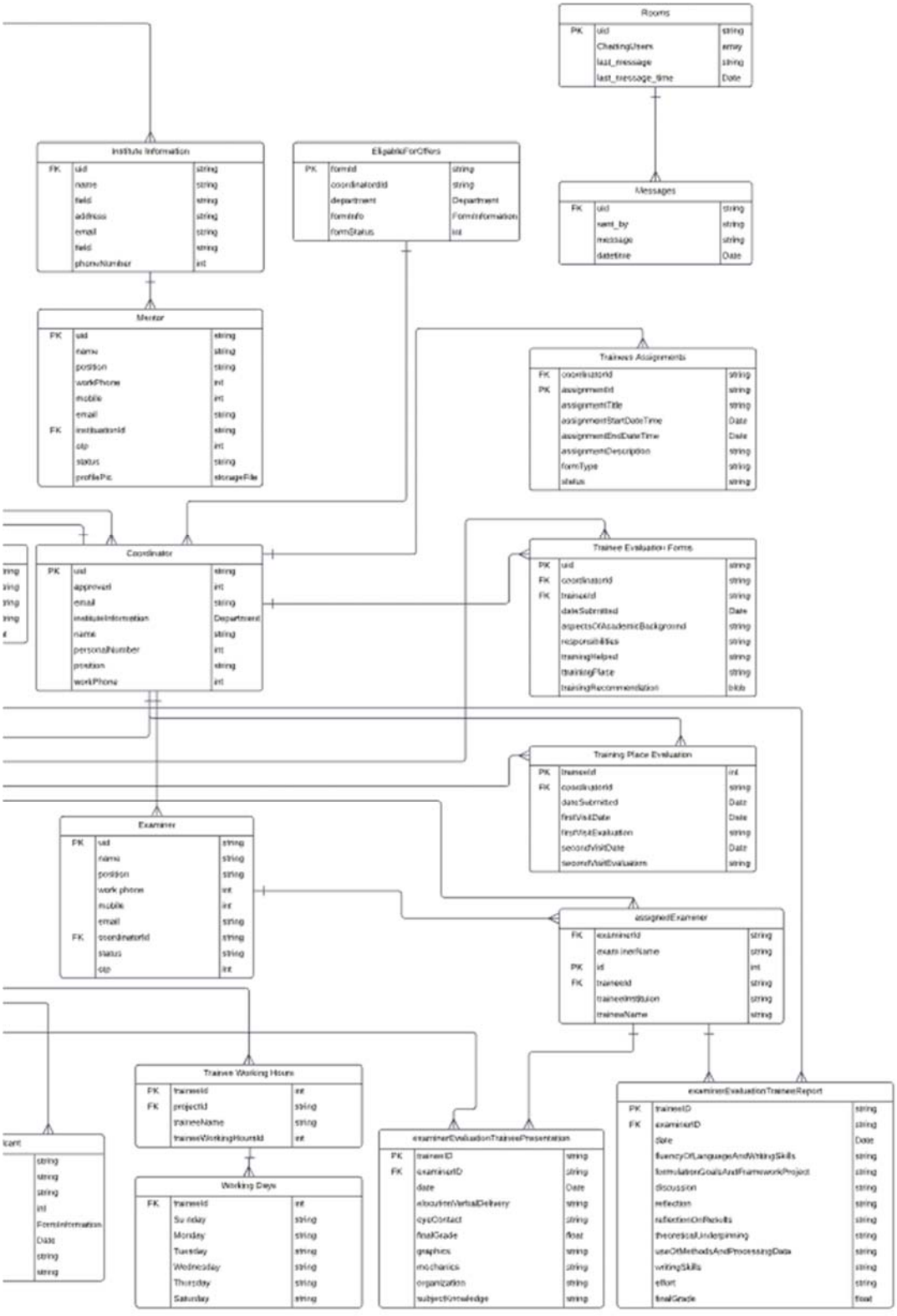


Figure 80. Database Diagram (2)

5.5. User interface design

In line with our software development process, our coding phase was carried out with utmost care. We ensured high-quality, efficient code that aligns with our defined software requirements.

The overall layout and visual design of the user interface is simple and universal, with a color scheme that combines light blue and white. This palette creates a clean, professional look that appeals to all user groups. Typography has been carefully chosen to ensure readability and maintain a polished appearance. Additionally, graphics have been employed to provide visual cues and enhance user experience.

5.5.1: Log-in and Registration

Upon launching the app for the first time, users are presented with a screen offering login as shown in figure 83, sign-up, and password recovery options. When signing up, users can choose their role as a Trainee, Employer, or Department, which determines the forms they need to complete. Following successful registration, a confirmation message is sent to the user's email address. The following figures 81, 82 and 84 display the registration process.

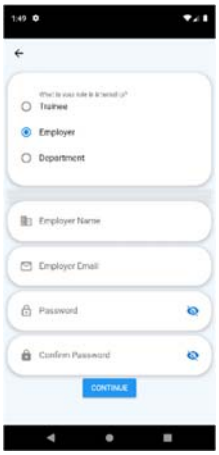


Figure 82. Initial Registration

Figure 83. Login Page

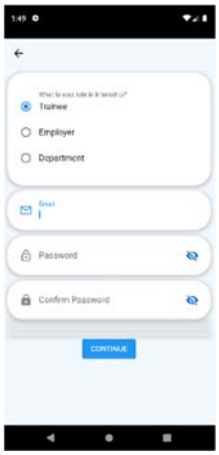


Figure 84. Account types

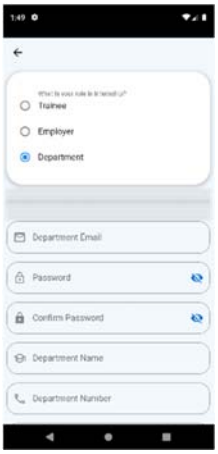


Figure 81. Full Registration

5.5.2: Coordinator View

On the Coordinator page, accessible after signing up as a department and receiving admin confirmation, the coordinator can view and edit their profile information as shown in figure 88, as well as manage Examiners (figure 87). The Examiners are authorized to fill out grading forms for Trainees. The coordinator page consists of several sections, each serving a different purpose. Figure 86 shows that the examiner can be registered through the coordinator and as in figure 85 his profile can be displayed to whoever registered him.

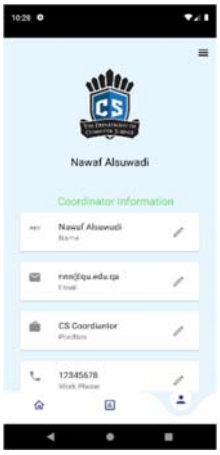


Figure 88. Coordinator Profile

Figure 87. Coordinator's Examiners

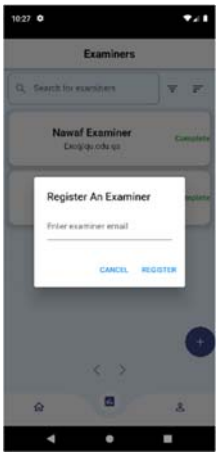


Figure 86. Register an Examiner

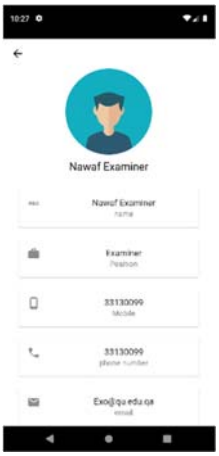


Figure 85. Coordinator View Examiners Profile

In the Employer section, the coordinator can view employer information and ensure their details are up to date. The Offer Management section allows the coordinator to send offer forms to employers, who will then provide information about available internships. Upon completion, the coordinator can review the submitted offer forms and send project forms with additional details for the employees to complete. Once finalized, the coordinator can publish the vacancies to Trainees within their organization, identifiable by the same email domain. Figure 89 is the coordinator homepage; from there he can go to the employer's page that is figure 90 or display initial forms as in figure 91 and click on any of them to view its details (figure 92).

Figure 89. Employer Details

Figure 91. Received Initial Forms From Employers

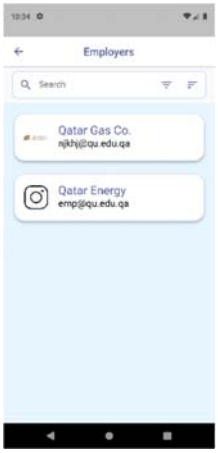


Figure 92. Current Available Employers

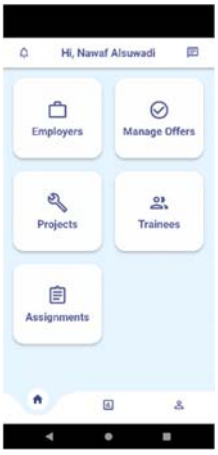


Figure 90. Coordinators Home Page

The coordinator can also create various types of assignments tailored to the Trainees in their organization through the Assignment Creation section (figure 96). In the Trainee Management section, when the coordinator clicks on a specific Trainee, they are directed to a detailed Trainee page (figure 97). This page includes the Trainee profile and a few forms related to that Trainee. The coordinator can also access the Trainees' projects and the tasks that the Mentor has assigned to them (figure 93). Both figures 99 and 100 are forms that the coordinator needs to fill up during the internship, they are related to each specific trainee.

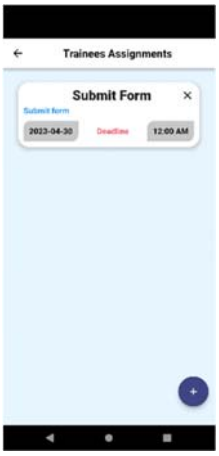


Figure 95. My Trainees Assignments

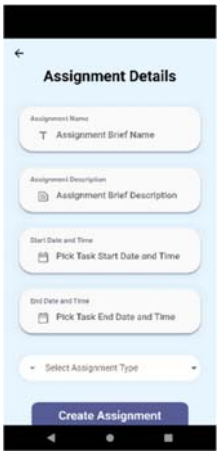


Figure 94. Assignment Details

Figure 96. My Trainees Projects

Figure 93. Mentor's Project View

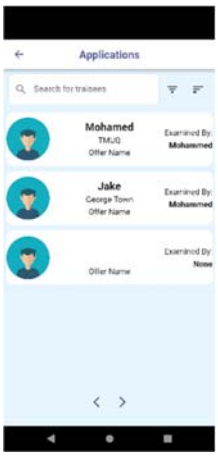


Figure 98. Trainee List

Figure 97. Trainees Action Menu

Figure 99. Attendance Table

Figure 100. Trainee Evaluation Form

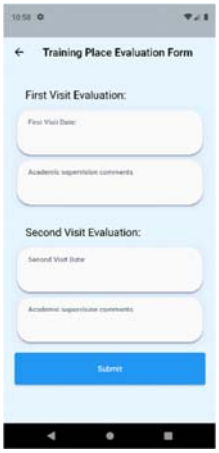


Figure 101. Training Place Evaluation Form

5.5.3: Employer View

Upon registering as an Employer and receiving admin confirmation, users gain access to the Employer page. Here, they can examine and modify their profile information (figure 102), view and add Mentors responsible for guiding Trainees' development as displayed in both figure 103 and 104 and create projects.

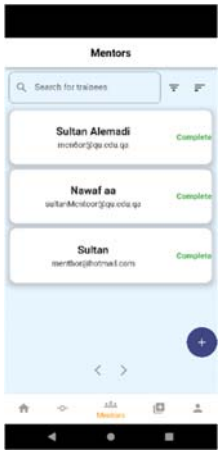


Figure 104. All Mentors List

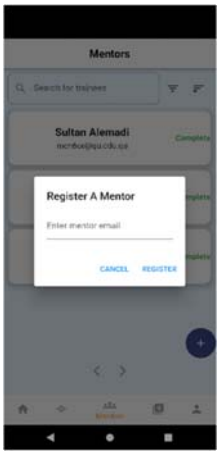


Figure 103. Add A Mentor

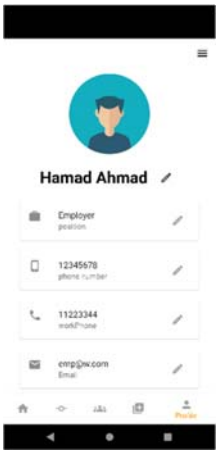


Figure 102. Employer/Focal point Profile

In figure 106, the Employer page also features a communications section with two tabs. The first tab displays application forms sent by Coordinators who are interested in receiving project offers from the Employer. Figure 107 shows that Employers can fill out and submit these forms to provide initial project information. If a Coordinator approves the submitted form, a confirmation form in figure 105 with additional project details is sent to the Employer for completion.

Figure 105. Employer Communication

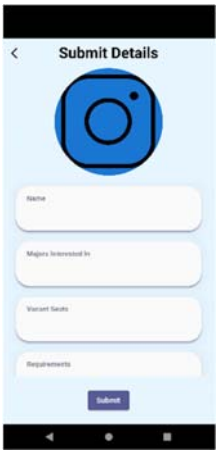


Figure 108. Initial Form

Figure 107. Confirmation Form

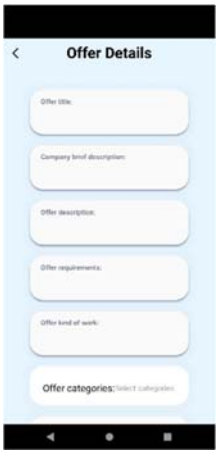


Figure 106. Offer Details

Once the Employer has filled out the offer details form, the coordinator publishes the offer. The published offer appears on the Employer's home page, displaying the offer title, description, posted date, and remaining seats (figure 109). Clicking on the offer directs the user to the offer details page (figure 110), which is divided into two tabs: one for the offer information and another for applications. In the applications tab, the Employer can review the profiles of Trainees who have applied for the offer and decide whether to accept or reject their applications.

Figure 109. Employer Home Page

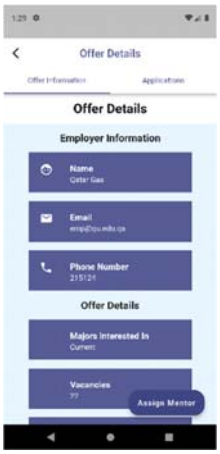


Figure 110. Offer Details

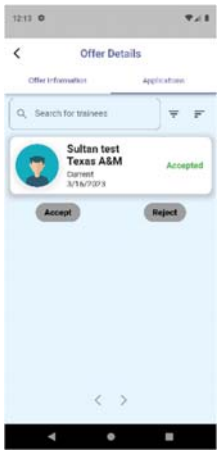


Figure 111. Offer Applicants

Figure 112. Navigating to Trainees Profile

5.5.4: trainee View

Trainees can view current open offers and check their details as seen in figures 113 and 114. They can apply for an offer by writing a statement to support their application figure (115).

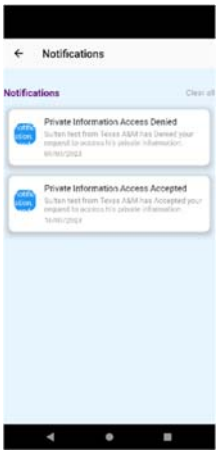


Figure 116. Notifications

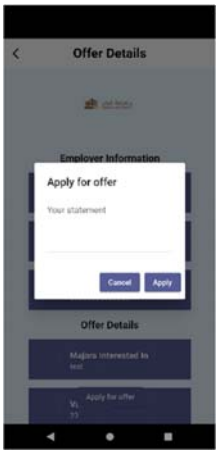


Figure 115. Offer Statement

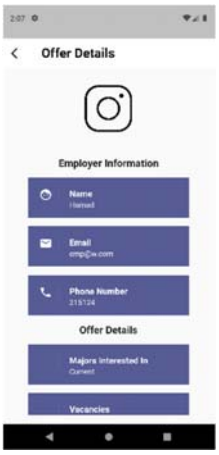


Figure 114. Offer Details (2)

Figure 113. Published Offers

Once a Trainee is accepted for an offer by the Employer, they can see tasks and assignments assigned by the Mentor or Coordinator, sorted by date figure (117). As tasks are completed, Trainees can mark them as done, figure 118 shows. Additionally, Trainees can view and edit their information on their profile page figure (119).

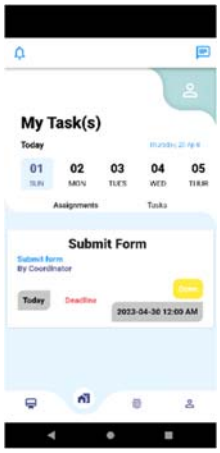


Figure 117. Trainee's Assignments

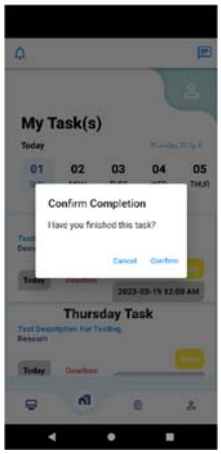


Figure 118. Finishing from a task

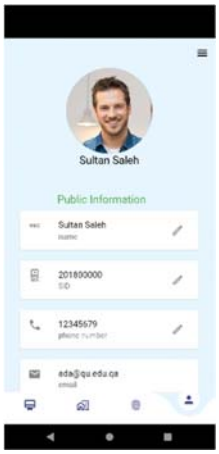


Figure 119. Trainees Profile

5.5.5: Mentor View

Mentors can view and create projects (figure 122) for the Trainees associated with the offer they have been assigned by the Employers. By clicking on a project, Mentors can view, delete, or create new tasks for Trainees displayed in figure 121.

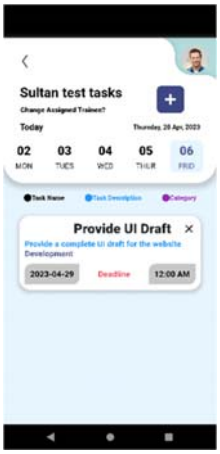


Figure 123. Mentor Project Screen

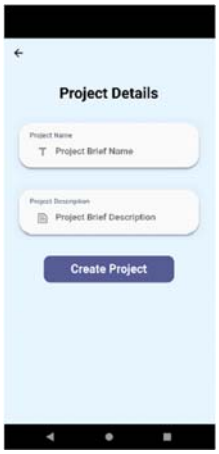


Figure 122. Project Details

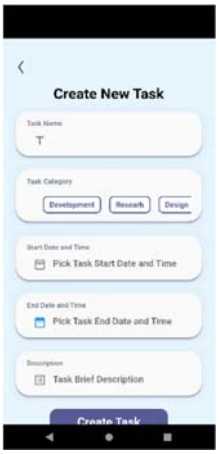


Figure 121. Mentor Add Tasks

Figure 120. Mentor All Projects view

They can also submit attendance hours for Trainees as the project progresses figure (124). Mentors can view the offer they have been assigned figure (125) to and see both the Employer's information and offer details figure (126). Furthermore, as seen in figure 127 mentors can view and modify their information on their profile page.

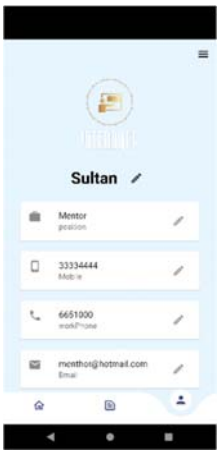


Figure 127. Mentors Profile

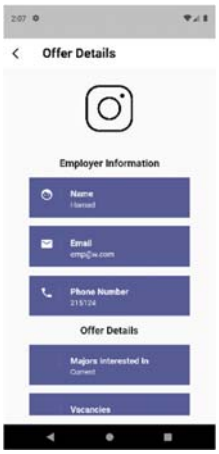


Figure 126. Offers Details (3)

Figure 125. Mentor's Offers

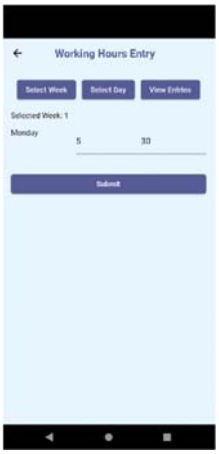


Figure 124. Weekly Hours Entry

5.5.6: Examiner View

Examiners, who are assigned by the coordinator to evaluate Trainees, can view the Trainees they are responsible for assessing (figure 128). By clicking on a Trainee's card, Examiners can review reports and presentations submitted by the Trainee (figure 129). They can grade and evaluate these materials by clicking on the pin icon in the top right corner of the tab or view the submitted PDF or Word files by clicking on the icon in the top left corner (figure 130). Examiners also could view and modify their information on their profile page figure (131).

Figure 128. Examiners Profile

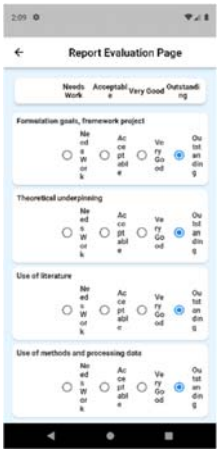


Figure 129. Report Evaluation Page

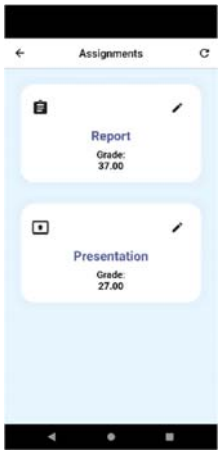


Figure 131. Trainees Assignments

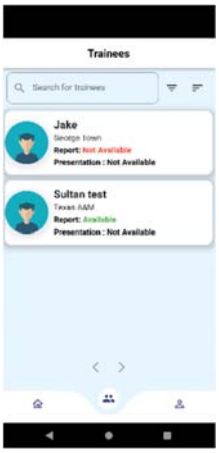


Figure 130. Trainees Under Examination

5.6. Design patterns

The application of the MVVM pattern in the project significantly influences the system's quality attributes. Here's a detailed evaluation:

Maintainability

The separation of concerns inherent in the MVVM pattern enhances maintainability. With each component (Model, View, View Model) encapsulating a distinct functionality, it's easier to update one part without affecting the others. This compartmentalization enhances navigability within complex systems and simplifies their upkeep, as demonstrated in Figure 132.

Scalability

MVVM pattern benefits scalability due to its modular structure. As the system grows, new Views can be added with their own dedicated View Models, which can access and share Models as needed. This architecture makes it easier to keep it organized and manageable.

Reusability

The MVVM pattern promotes reusability, especially of the View Model and Model components. This allows for faster development and potential code reduction when developing new features or interfaces. However, care must be taken to design these components with reuse in mind to avoid tight coupling that could limit their applicability.

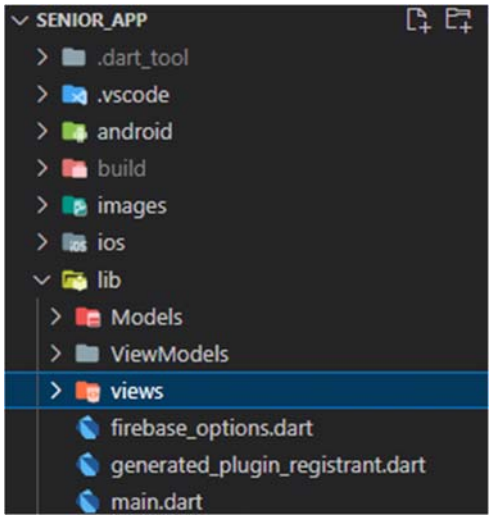


Figure 132. MVVM - VSCode

Design Rationale

1. Responsive UI Requirements: For the web application, there was a need for a responsive UI that could reflect changes in data in real-time. MVVM, with its robust data-binding capabilities, fits this requirement perfectly.
2. Project Complexity and Future Growth: The project was not a simple application with a static UI. It was expected to grow in complexity over time, necessitating a design pattern that supports modularity and scalability. The separation of concerns in MVVM, with distinct Model, View, and View Model components, made it an ideal choice for managing this complexity and supporting future expansion.

3. Focus on User Interaction: The application had a significant emphasis on user interaction. MVVM is well-suited to user-centric applications as it ensures UI updates (via the View) are in sync with data changes (in the Model), managed by the View Model.

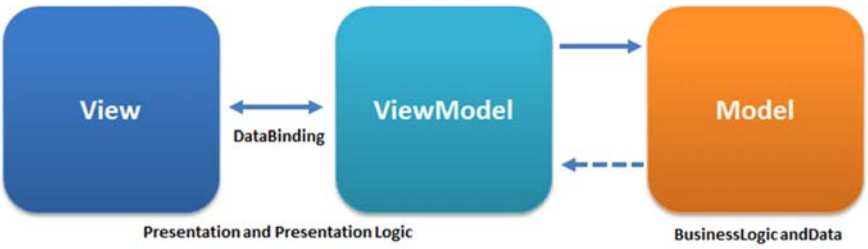


Figure 133. MVVM Logic

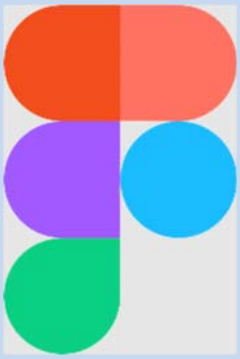
6. Implementation

6.1. Hardware/software used

As part of our ongoing software development process, we evaluated various potential solutions and selected the most fitting ones that align seamlessly with our software requirements. This deliberate selection process involved comprehensive analysis and comparison of different technologies and tools, ensuring we adopted the best available options. Our decision-making was influenced by many factors such as ease of use, scalability, cost, and compatibility with existing systems, all aiming to optimize the functionality and performance of our final product.

Table 10. Hardware/ Software to be used

Platforms/Tools **required to develop the software**	HW/SW details	Justifications
	Flutter is an open-source framework for developing top-notch, high-performing mobile and web applications for Android, iOS, Windows, MacOS, and Linux. It offers a straightforward, strong, effective, and simple SDK for creating applications in Dart, Google's own programming language.	We chose flutter because it supports a diversity of platforms. Within one codebase, we can build the same application for different platforms! This is a huge advantage because we will not have to write more than one code.
	Visual Studio Code is a supporter for debugging, task execution, and version control by a simplified code editor. It tries to give developers only the tools they require for a short cycle of code-build-debugging and leaves more sophisticated processes to IDEs with more features, like Visual Studio IDE.	The choice falls in the visual studio as IDE because it supports a lot of programming languages including the language we will work on. Additionally, the provided syntax highlighting, bracket matching, auto-indentation, box selection, and snippets will help us become more productive right away.
	Dart is an object-oriented language with C-style syntax. Unlike other programming languages, it offers programming notions like interfaces and classes. Arrays are not supported by Dart. The replication of data structures like arrays, generics, and optional type is possible with Dart collections.	In order to work with Flutter and achieve our goal, which is to write one codebase for all different platforms, the Dart language was the only and best option.

	Firebase provides hosting services for different kinds of applications like Android, iOS, JavaScript, Node.js, Java, C++, etc. It provides real-time and NoSQL hosting for databases, content, social authentication), alerts, and other services like a real-time communication server.	The selection of the database was not easy because there were many options. Our selection falls on firebase because of the many advantages it has. The most important advantages are the real-time data, ready-made API, auto create diagram option, the security is built-in in nodes so no worry about the security, the file backup is by Google Cloud, the hosting file is static, the data is treated as streams, the ability of build a highly scalable database, the usability.
	One of the most innovative graphics editing tools, Figma, is sweeping the design industry. The fact that it is cost-free to use is what makes it so alluring.	One of the best applications for sketching. It has many advantages that make him first in the race. The most important advantages include real-time collaboration, browser-based, easy handoff, layout grid, Free, and easy to learn.
	GitHub is a platform for version control and team collaboration that hosts code. It enables remote collaboration on projects between you and other people. There are features in GitHub like repositories, branches, commits, and pull requests in this lesson.	We chose to use GitHub because it is the easiest way to work in an implementation of shared code and files. Also, because it is free to use.

6.2. Challenging issues and solutions

In our project, we capitalized on the synergies of Flutter and Dart for our front-end and Firebase for our backend. These strategic choices have been instrumental in realizing the project's potential. We initiated Firebase setup through a web browser, following a series of steps to integrate it with our base code.

Our data structure utilized Firebase's collections and documents, established in model classes. These classes supported a multitude of data types, including strings, integers, dates, Booleans, and arrays. Within each model class, a mapping function efficiently linked these attributes to the Firebase database.

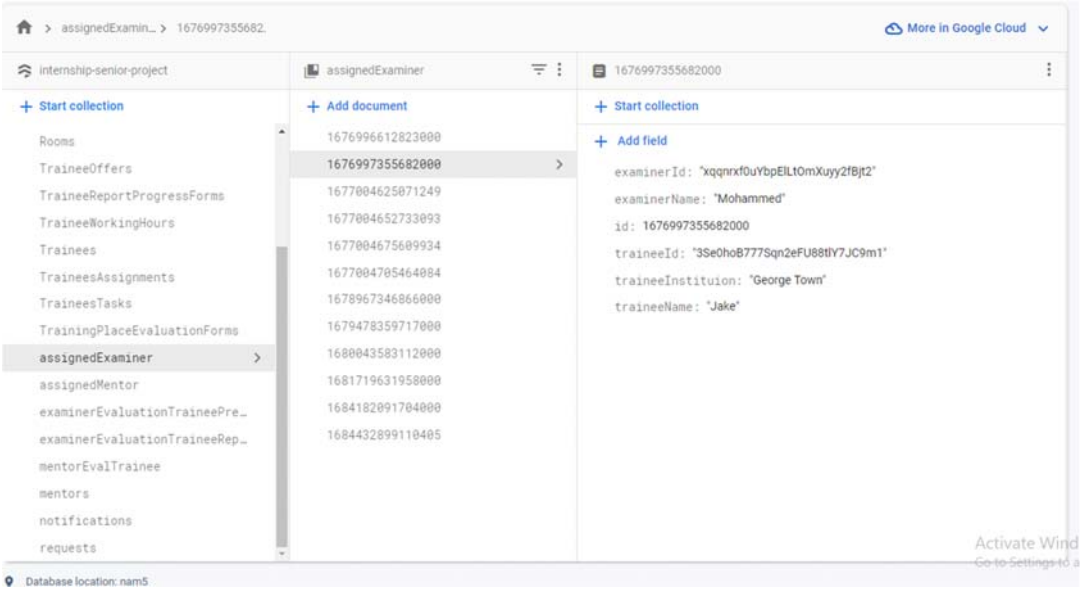


Figure 134. Firebase Collection

```dart
class assignedExaminer {
  int? id;
  String? traineeId;
  String? traineeName;
  String? traineeInstituion;
  String? examinerId;
  String? examinerName;

  assignedExaminer(
      {this.id,
      this.traineeId,
      this.traineeName,
      this.traineeInstituion,
      this.examinerId,
      this.examinerName});
  factory assignedExaminer.fromJson(Map<String, dynamic> json) {
    return assignedExaminer(
        id: json['id'],
        traineeId: json['traineeId'],
        traineeName: json['traineeName'],
        traineeInstituion: json['traineeInstituion'],
        examinerId: json['examinerId'],
        examinerName: json['examinerName']);
  }
  Map<String, dynamic> toMap() {
    return {
      "id": id,
      "traineeId": traineeId,
      "traineeName": traineeName,
      "traineeInstituion": traineeInstituion,
      "examinerId": examinerId,
      "examinerName": examinerName
    };
  }
```

Figure 135. Model Class

Our front-end, designed with Dart and Flutter, handled the reading and writing operations. These powerful languages, renowned for their ease of use and speed of development, facilitated a seamless interaction with the backend. Moreover, the combination of Flutter's rich widget library enabled us to create an intuitive and user-friendly interface.

Leveraging Firebase's features considerably streamlined our coding process. With its simplicity, tasks like updating, adding, and deleting became a matter of a few code lines. Notably, Firebase's filtering capability shone when retrieving information based on the user's ID, an operation that our system had to perform repeatedly.

```dart
import 'package:get/get.dart';
import 'package:senior_app/Models/approveRequestModel.dart';

class approveRequest_ViewModel extends GetxController {
  var allRequests = <approveRequestModel>[].obs;
  @override
  void onInit() {
    super.onInit;
    //fetchAllRequests();
  }

  Stream<List<approveRequestModel>> readAll() => FirebaseFirestore.instance
      .collection("requests")
      .snapshots()
      .map((snapshot) => snapshot.docs
          .map((doc) => approveRequestModel.fromJson(doc.data()))
          .toList());

  Stream<List<approveRequestModel>> readAllPending(String traineeUid) {
    return FirebaseFirestore.instance
        .collection("requests")
        .where("status", isEqualTo: "Pending")
        .where("traineeId", isEqualTo: traineeUid)
        .snapshots()
        .map((snapshot) => snapshot.docs
            .map((doc) => approveRequestModel.fromJson(doc.data()))
            .toList());
  }

  Future<approveRequestModel?> getRequest(
      String traineeId, String empId) async {
    var requestRef = FirebaseFirestore.instance
        .collection("requests")
        .where("traineeId", isEqualTo: traineeId)
        .where("employerId", isEqualTo: empId);
    var requestSnapshot = await requestRef.get();
    if (requestSnapshot.docs.isNotEmpty) {
      return approveRequestModel.fromJson(requestSnapshot.docs.first.data());
    } else {
```

Figure 136. View-Model Class

The integration of Firebase, Dart, and Flutter set the stage for an efficient, responsive, and user-centric application, significantly enhancing the overall experience of the users.

Throughout the implementation of our project, we encountered various challenges. By addressing these challenges, we gained valuable insights that may benefit others working on similar projects. Some of the key challenges faced and their respective solutions are discussed below:

- Challenge 1: Understanding how firebase works.
  Solution: Our initial encounters with Firebase were complex due to its unique behavior, resulting in some functions not performing as expected. This required substantial time and research for us to adapt. One particular issue concerned the registration process; when an employer registered a mentor or a coordinator registered an examiner, they were inadvertently logged into the new accounts. Further complicating matters, there was a risk of becoming indefinitely trapped in the "Complete registration" screen, requiring database access for resolution.

- Challenge 2: Managing Firebase DB security rules and access control.

Solution: Ensuring the security and privacy of user data in Firebase DB was critical. We carefully designed our database security rules to limit access to authorized users and restrict read/write permissions based on user roles.

- Challenge 3: Dealing with platform-specific issues in Flutter.
  Solution: One of the advantages of using Flutter is its ability to create cross-platform applications. However, this can also introduce platform-specific challenges. To tackle these issues, we thoroughly tested our application on both iOS and Android devices, identifying and resolving platform-specific quirks and differences. We also leveraged Flutter's platform-specific APIs and plugins when necessary to provide a consistent user experience across platforms.

- Challenge 4: Managing both senior project and other university courses.
  Solution: Effective time management and prioritization were essential in balancing the workload between the senior project and other university courses. We established a clear schedule, setting milestones and deadlines for each phase of the project, while allocating sufficient time for coursework and other academic responsibilities. Regular meetings and open communication among team members also ensured that tasks were distributed evenly, and progress was monitored consistently.

- Challenge 5: Learning a new programming language - Dart/Flutter
  Solution: To overcome the challenge of learning Dart/Flutter, our team dedicated time for self-study, utilizing online resources and tutorials. By sharing knowledge and helping one another, our team was able to develop the necessary skills and become proficient in Dart/Flutter.

- Challenge 6: The project kept evolving in terms of size.
  Solution: As the project progressed, its scope and size expanded, demanding more time. To address this, we continually revisited our project plan, adjusting timelines and resource allocations as needed. We also prioritized features, focusing on those that were most crucial to the project's success. This agile approach allowed us to adapt to changes and maintain steady progress towards our goals.

- Challenge 7: Ensuring effective communication among team members.
  Solution: To foster seamless collaboration, we held regular weekly meetings to discuss progress, address concerns, and share updates. This open and transparent approach helped to build trust and maintain momentum throughout the project.

7. Testing and evaluation

7.1. Functional testing

In Chapter 3 of our software development process, we incorporated both functional and non-functional testing as integral parts of our testing assurance strategy. Functional testing was conducted to ensure that the software behaved as expected, with each function being tested by providing appropriate input and verifying the output. This enabled us to validate that each requirement was correctly implemented. Our testing included two specific tests - Flutter's Unit/Integration test and Acceptance test, both of which were important in uncovering hidden bugs and errors. We found Flutter's Unit/integration tests particularly valuable and crucial, standing out as the most efficient tool in our testing, helping us ensure seamless component interactions within our software.

7.1.1 Unit & Integration Test

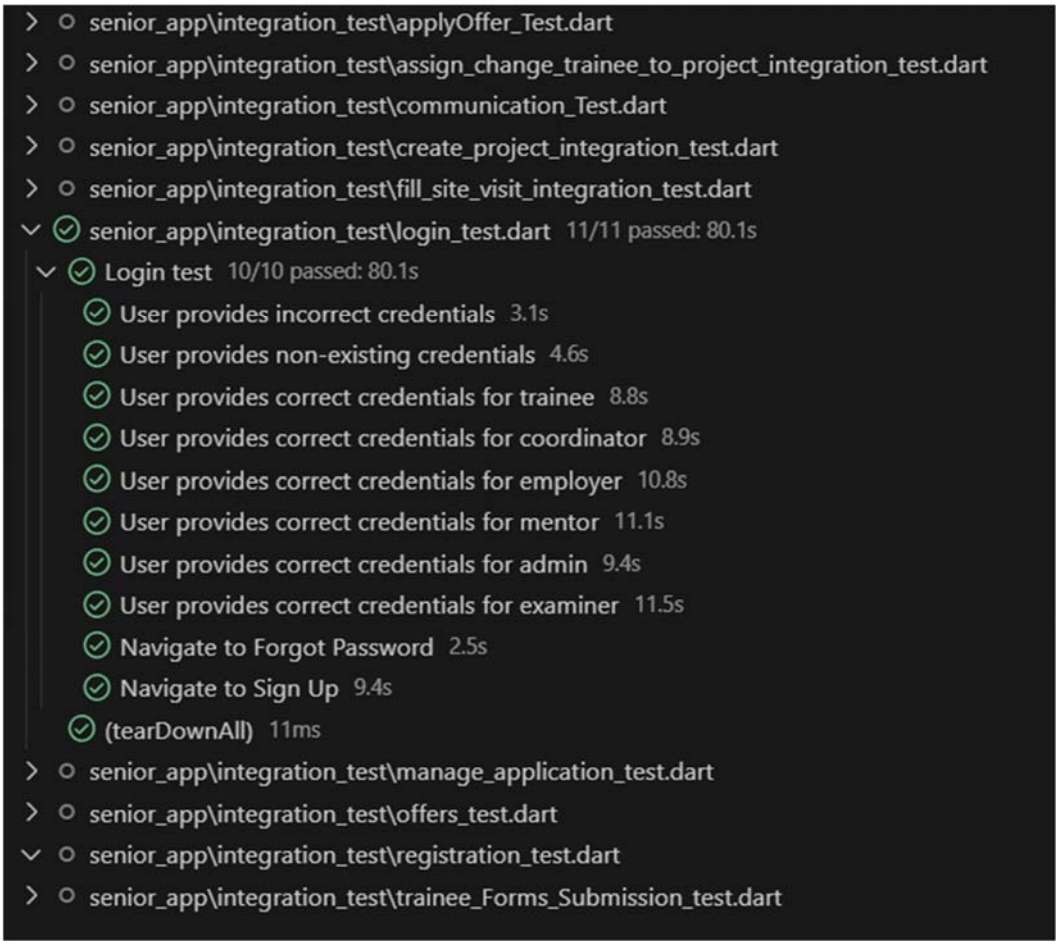


Figure 137. Login Testing

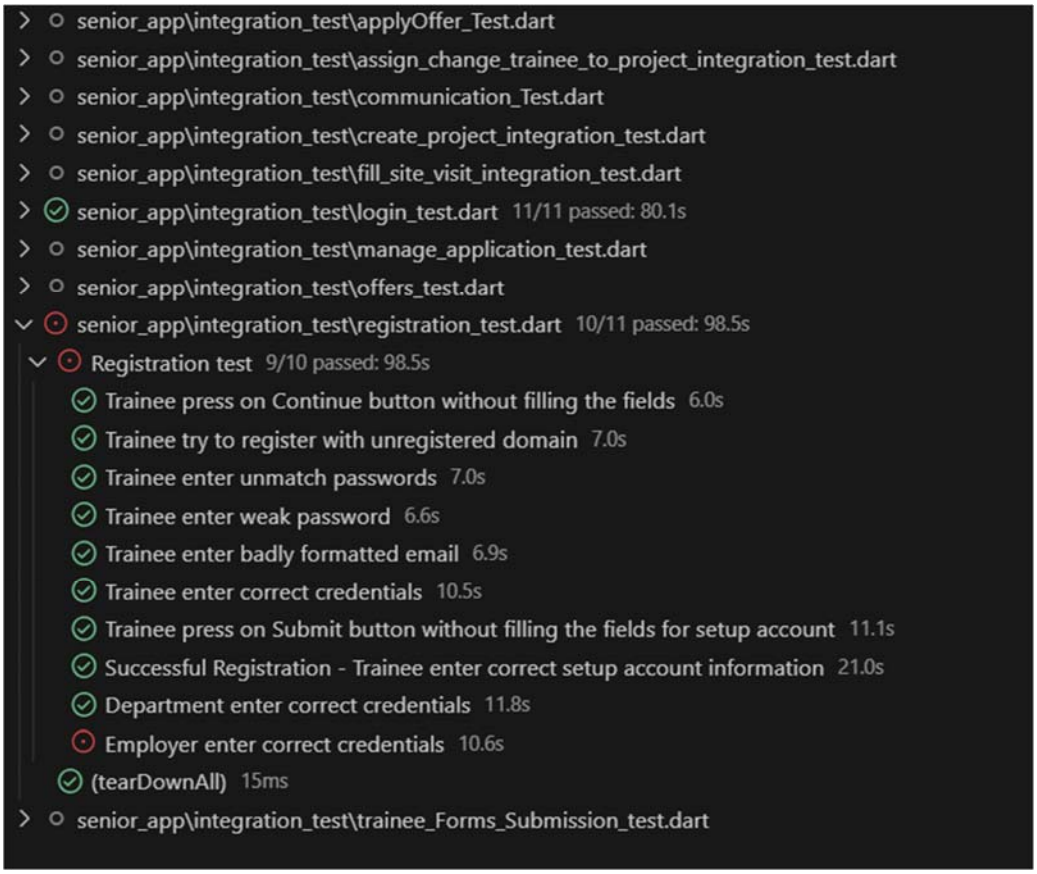


Figure 138. Registration Testing

All tests have successfully passed, except for the employer registration. While the employer registration functionality itself is functioning correctly, we have encountered an issue that is preventing the test from passing. Through thorough investigation and error searching, we have determined that the problem lies with the library and test widget we are currently utilizing.

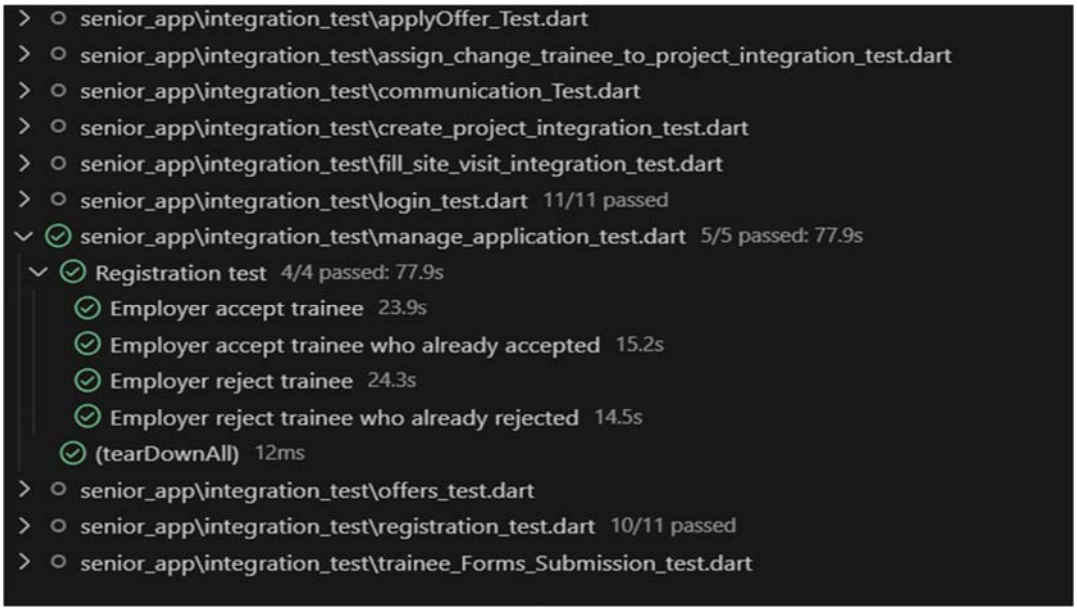


Figure 139. Manage Application Testing

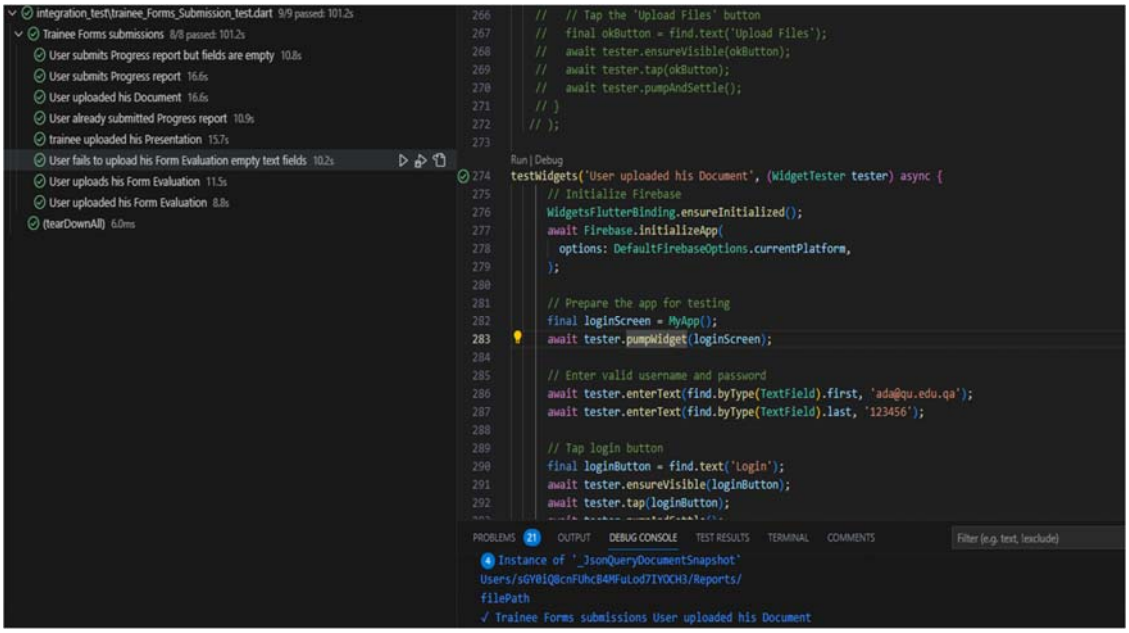


Figure 140. Trainee Forms Submissions Testing

Form Evaluation and Progress Report Tests: We implemented a series of integration tests on the form evaluation and progress report submissions functionality. The tests are designed to ensure both positive and negative validation, including the following scenarios:

- Successful submission: The test simulates the process of filling out the form with valid input and successfully submitting it.
- Validation of required fields: The test tries to submit the form with one or more required fields left empty and expects an error message to be displayed.
- Already submitted: The trainee already submitted his evaluation/feedback once.

2. **File Upload Tests**: Two separate tests were carried out on the file upload functionality.
    - Successful file upload: The test simulates the process of selecting a file and successfully uploading it.
    - Re-upload confirmation: The test simulates the process of uploading a file when a file has already been uploaded. The test expects a specific prompt to be displayed, asking the user to confirm whether they want to replace the existing file.

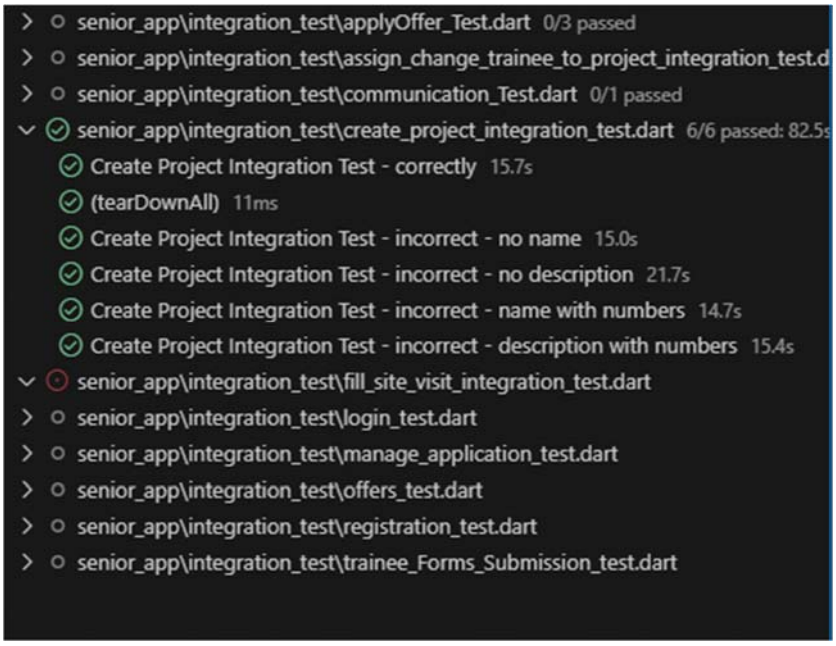


Figure 141. Create Project Testing

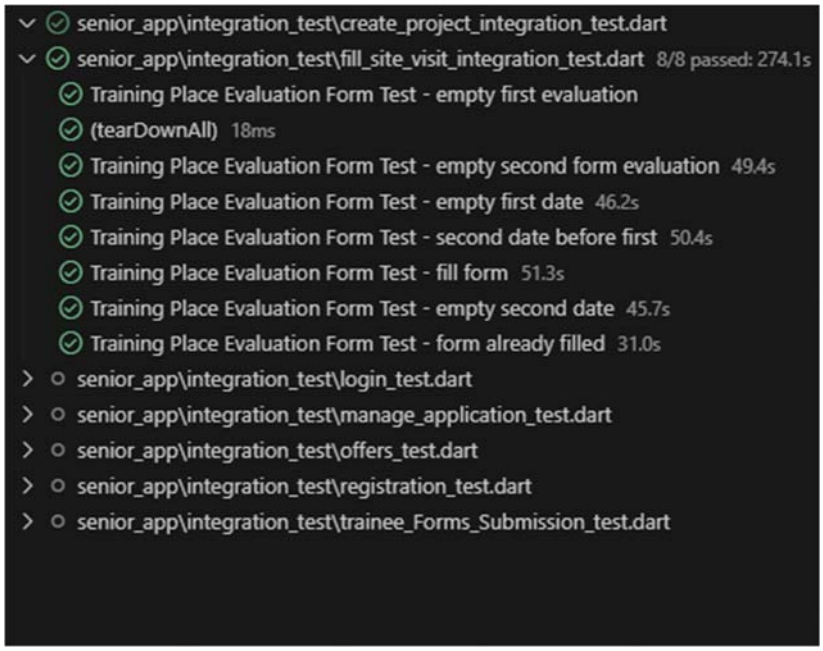


Figure 142. Fill site visit Testing

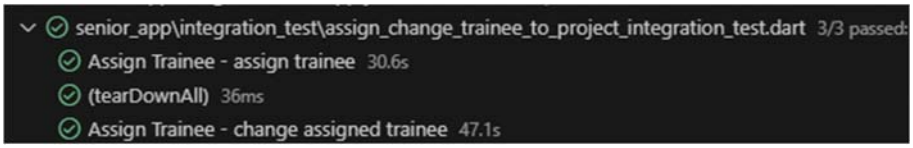


Figure 143. Assign Trainee Testing

7.1.2 Acceptance Test

In the acceptance testing phase, feedback from approximately 39 individuals provided insights into various facets of our application. Regarding user interface, a substantial 80% found it user-friendly and intuitive, suggesting a positive user experience. However, 20% pointed out that, while the interface was reasonably intuitive, there was room for refinement. This underlines the significance of persistently honing the interface to boost usability.

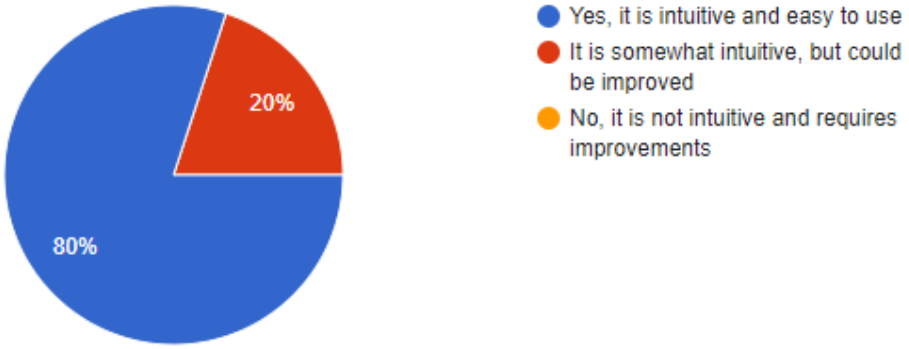


Figure 144. Acceptance Test - UI

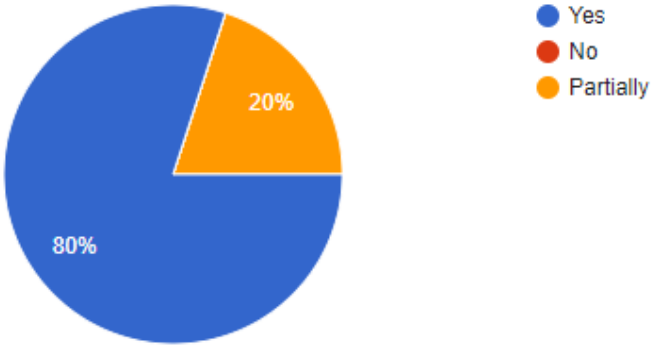


Figure 145. Acceptance Test - Expectation

When asked about whether the application fulfilled initial requirements and expectations, 80% of users affirmed that it met their needs, indicating its capability to satisfy user demands. However, a fifth of the users opined that it only partially lived up to their expectations. Some comments included expectations of more responsive design and desires for a cleaner, less detail-cluttered application.

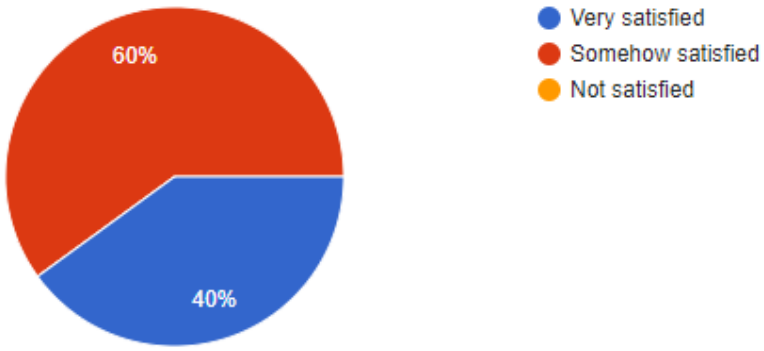


Figure 146. Acceptance Test - Performance

In terms of the overall performance and speed of the application, most users, 60%, considered it to be excellent, and 40% rated it as good. This feedback implies that the application performs optimally and is responsive, offering a smooth user experience. Despite this, it remains essential to take feedback from those users who graded the performance lower. More specific suggestions or examples of areas perceived as lacking or confusing would be beneficial.

7.1.3 System Test

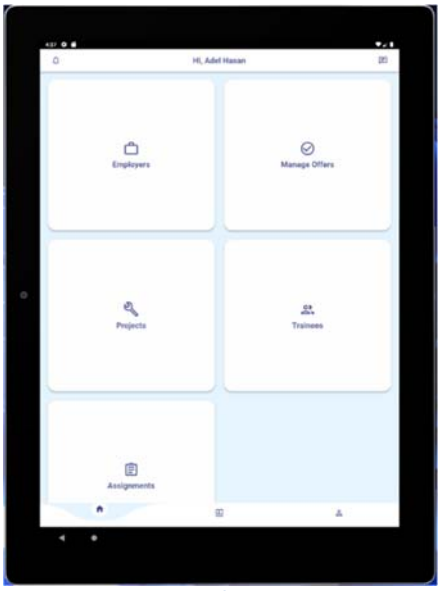


Figure 147. iPad Test

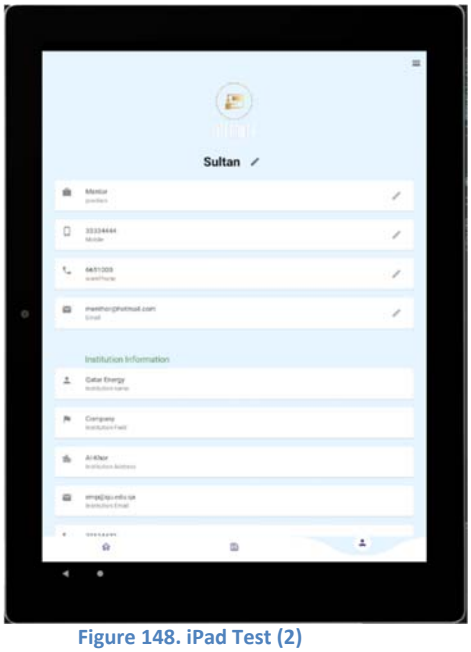


Figure 148. iPad Test (2)

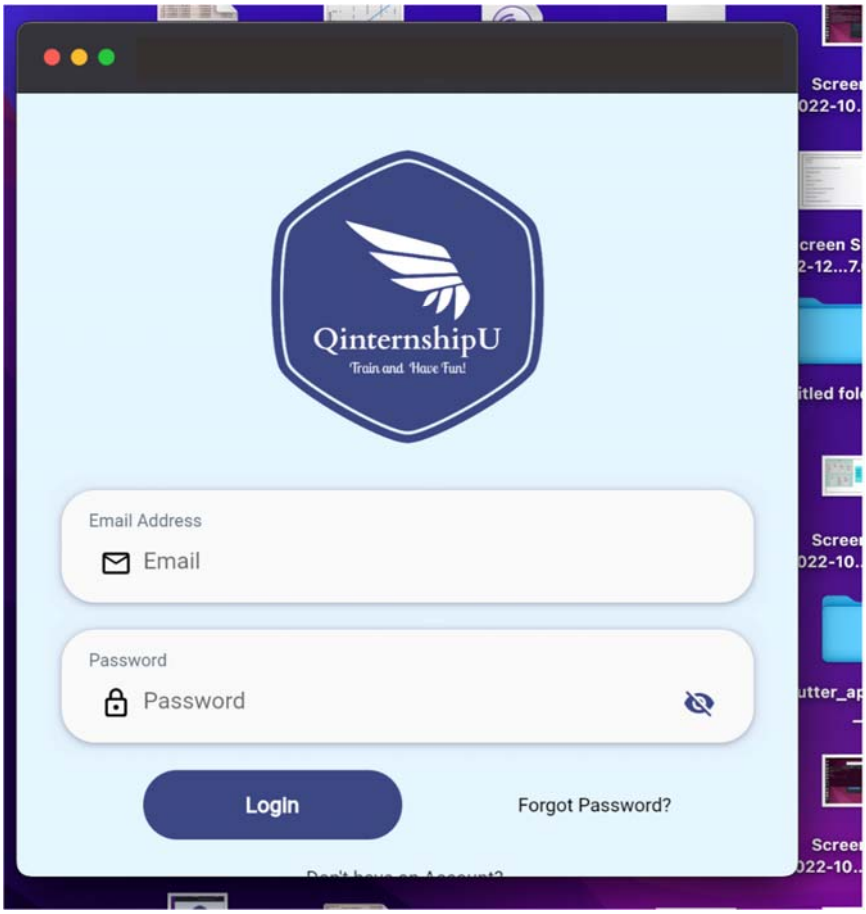


Figure 149. IOS System test

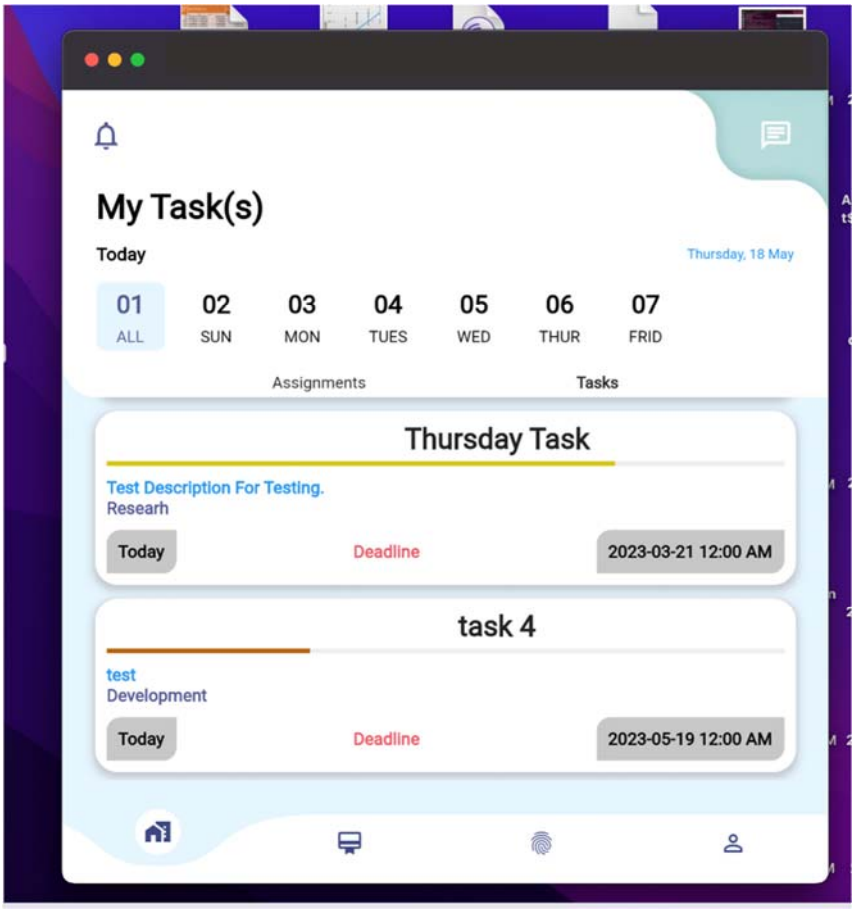


Figure 150. IOS System test (2)

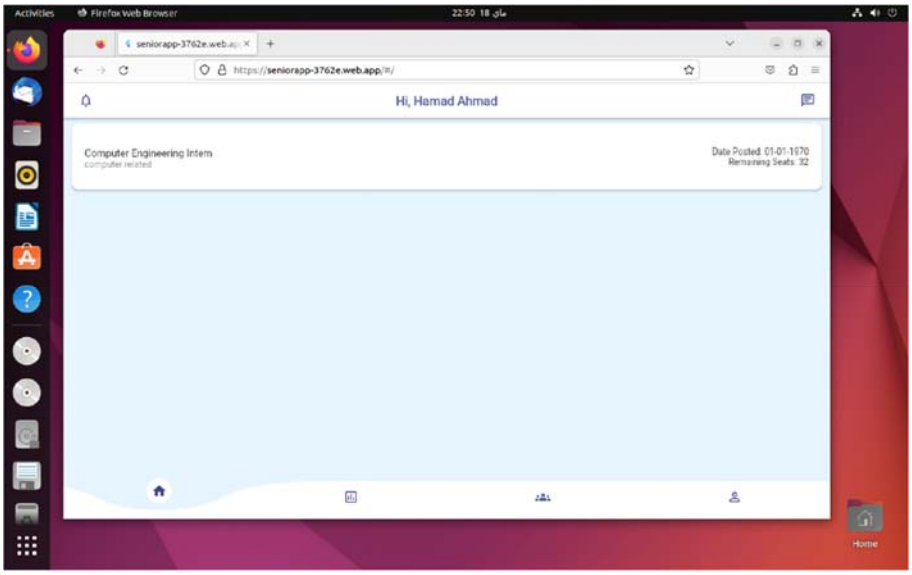


Figure 151. Linux System Test

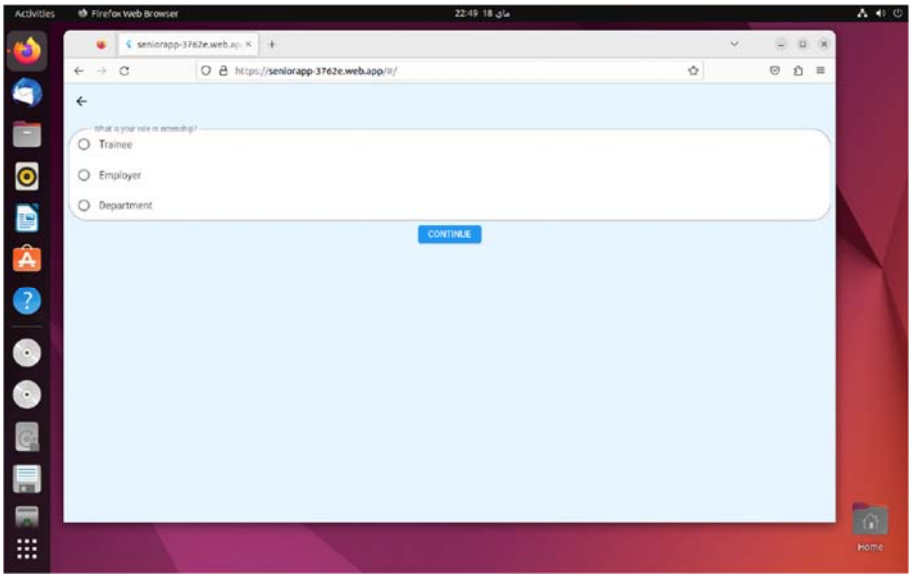


Figure 152. Linux System Test (2)

7.2. Non-functional testing

non-functional testing was performed to verify the system's performance and reliability under varying conditions. This included tests for scalability, availability, performance, security, usability, privacy, and portability. Through non-functional testing, we evaluated the software's readiness for real-world deployment and fine-tuned aspects like loading, and user-friendliness.

7.2.1 Privacy

By implementing this Privacy Policy, we are addressing and testing the non-functional requirement of privacy to ensure that user's private information is protected and only shared with their consent. The user will have to agree to our terms when registering, it will provide full confidentiality.

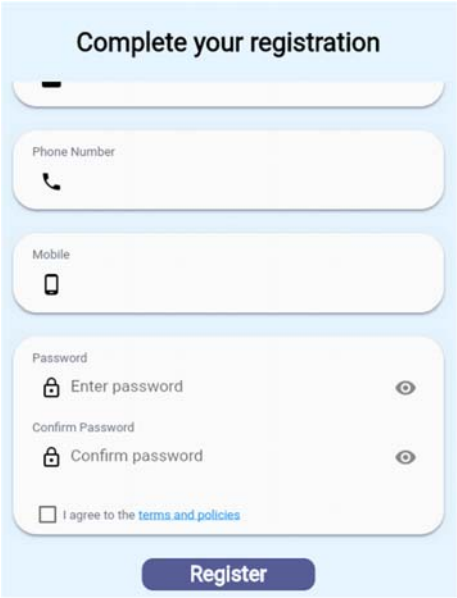


Figure 153. Privacy and Policy (1)

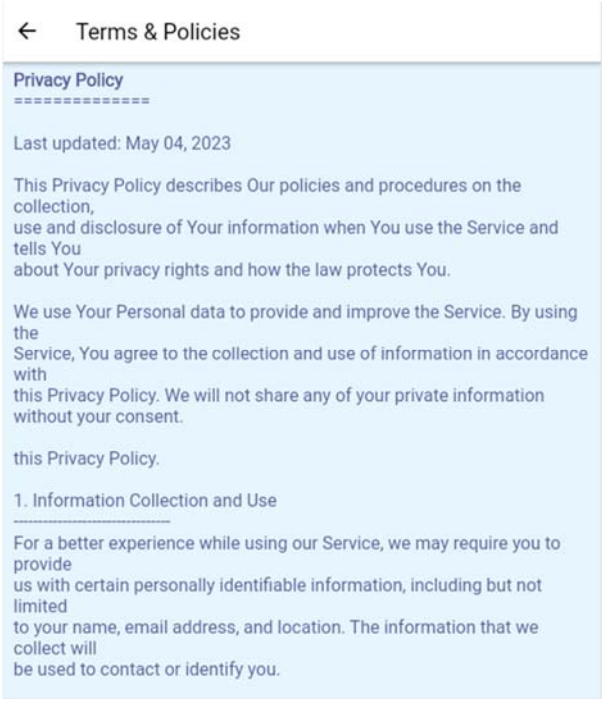


Figure 154. Privacy and Policy (2)

The system also includes a “Request Access” functionality that submits a request to the trainee, and if accepted the private information will be displayed to the focal point or the coordinator. If rejected, private information will not be accessed, and both will be notified on the trainee’s decision.

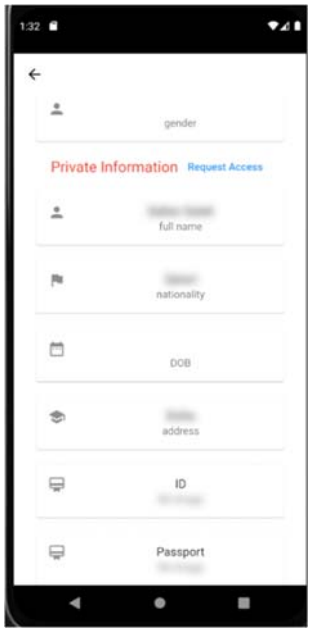


Figure 155. Access Request

7.2.2 Portability

In our application, portability is an advantage. This advantage came from using the Flutter framework. We can run the application on various platforms. For example, we can run the application on mobile phones, desktops, and web platforms. This advantage gives the user the choice to run the application on any platform he likes. In addition, it makes the user use the advantages of the platform he likes. For instance, if he chooses the mobile, he will benefit from using the app in any place he goes (the advantage of mobility). If he chooses to run the application on the desktop, then he will benefit from the speed and strength of his PC device (since desktops are faster and more responsive than the web). If the user chooses to run the application on the web, then he will benefit from using the application without needing to download and update it. The flexibility of the choice is particularly important for the customer to make him comfortable choosing which platform he likes to use its advantages. The following screenshots demonstrate the running of the application in mobile, desktop, and web for some screens in our application:

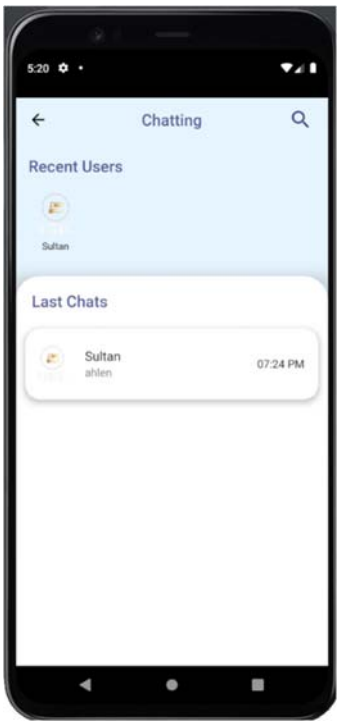


Figure 156. Portability Test on Phones (Android) (2)

Figure 157. Portability Test on Phones (Android)

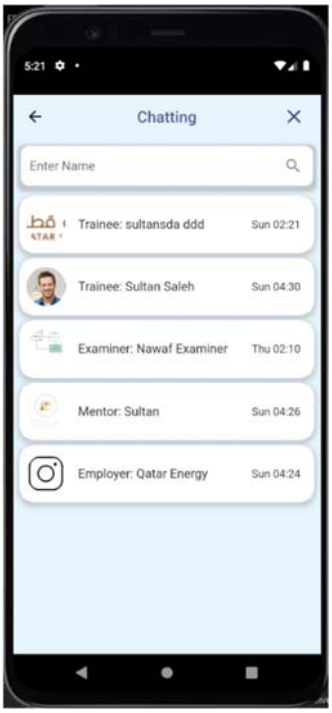


Figure 158. Portability Test on Phones (Android) (3)

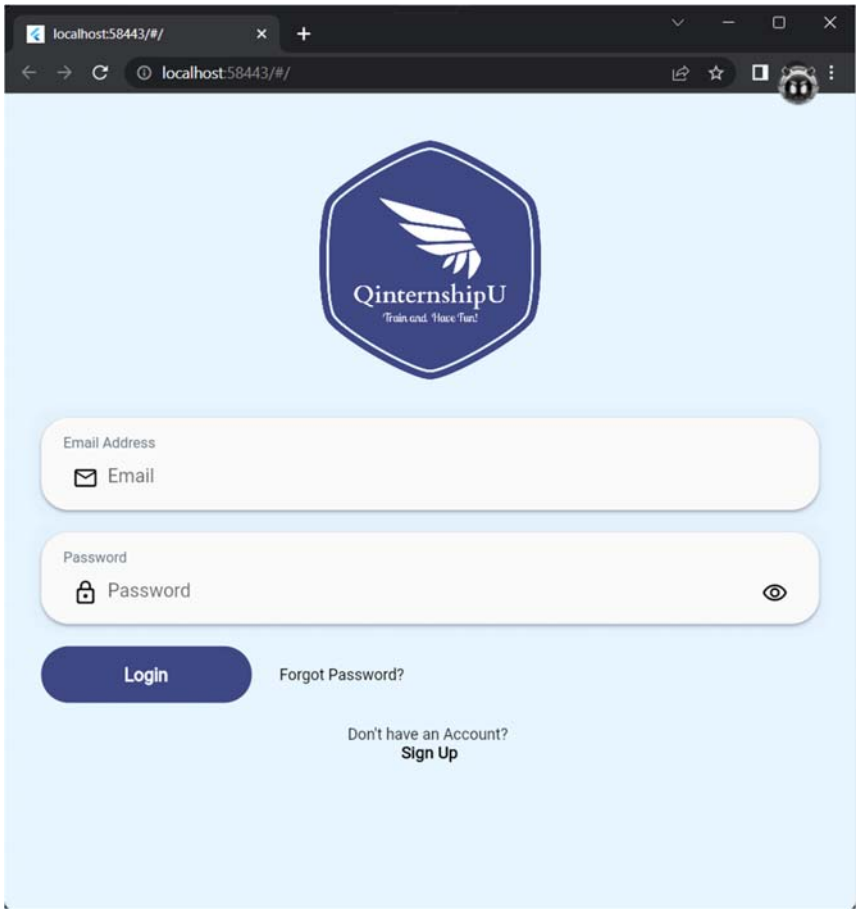


Figure 159. Portability Test on Web (1)

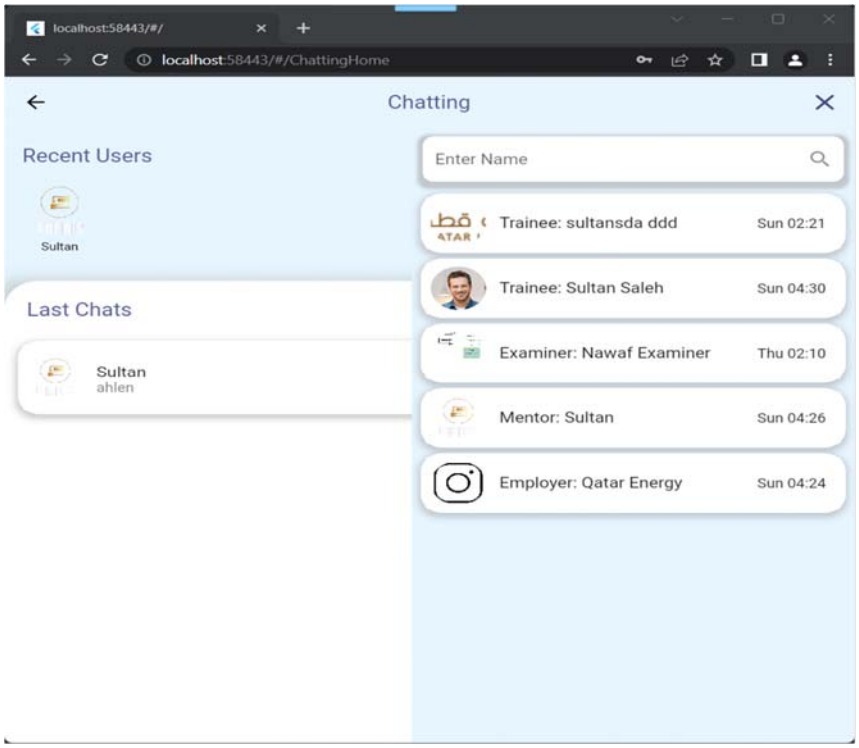


Figure 160. Portability Test on Web (2)

7.2.3 Performance

Efficient coding and optimized data processing are at the core of our application's lightning-fast performance. In fact, we were able to read 77 items in only 0.03 seconds, demonstrating our unwavering commitment to providing a smooth user experience. Even with a more complex query involving multiple related tables, the application still completed the task in an impressive 0.55 seconds. As we continue to improve and enhance the application, we will maintain our focus on delivering exceptional performance.

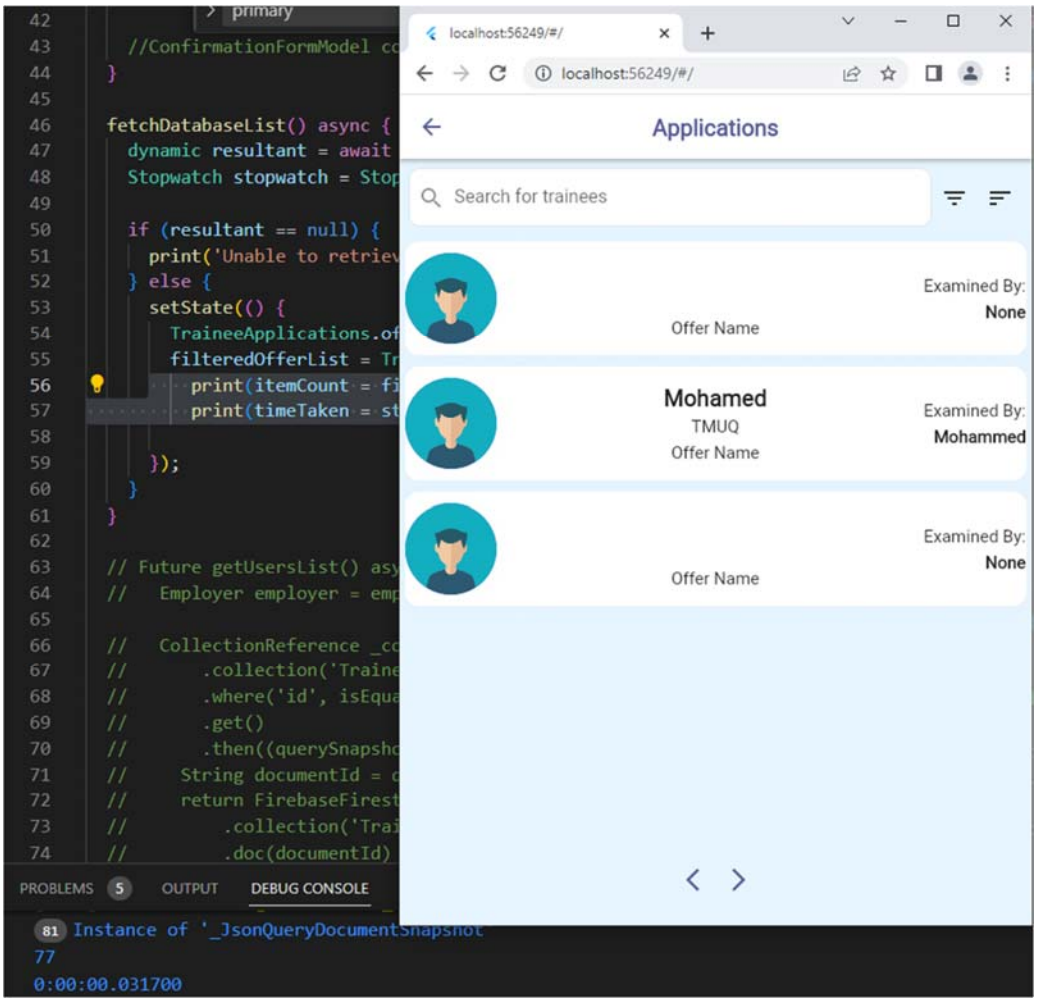


Figure 161. Performance Test (1)

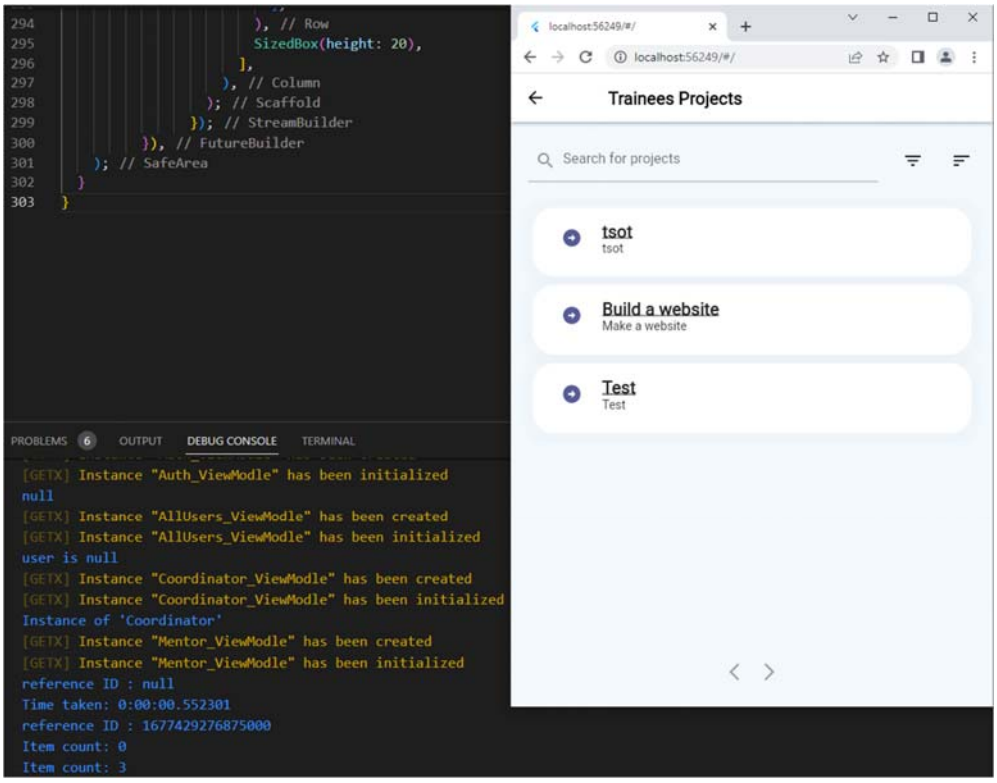


Figure 162. Performance Test (2)

7.2.4 Scalability

We utilized Locust, a Python-based open-source tool, for scalability testing. We executed three tests with escalating user numbers to assess the app's scalability: the first with 100 users, the second with 10,000, and the last with a significant leap to 100,000 users.

The initial test involved 100 users, and the application responded to requests within 100ms, indicating smooth functioning at this load level. The second test ramped up to 10,000 users, with the application still maintaining a consistent response time of around 1000ms.

In the final test involving 100,000 users, response times initially spiked but gradually reduced as maximum user load was reached, indicating the system could handle this maximum load albeit with increased response times.

Our tests demonstrated the application's ability to manage loads of up to 100,000 users, though response times tend to increase significantly at this level. This performance can be improved by scaling the application through code optimization, or load distribution using a load balancer.

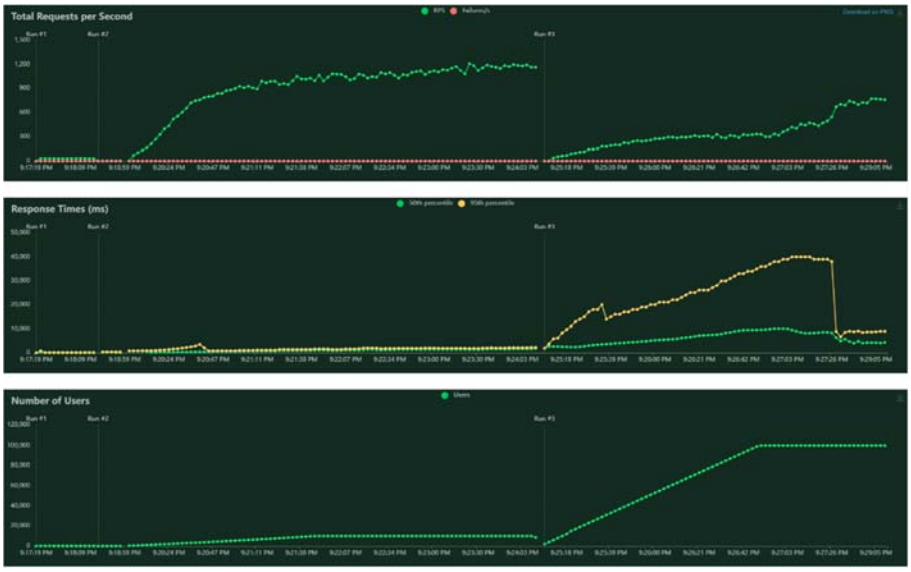


Figure 163. Scalability Test

7.2.5 Security

In our project, a robust set of security measures, a critical non-functional requirement, was incorporated to ensure data protection. Our backend service, Firebase, employs security rules by default, forcing authentication and data access validation. These rules are also customizable to fit specific security needs.

Figure 164. Firebase Rules

Firebase authentication, another key feature, securely stores passwords as hashes, thereby preventing unauthorized access. This service provides multiple user authentication methods and automatic handling of user sessions.

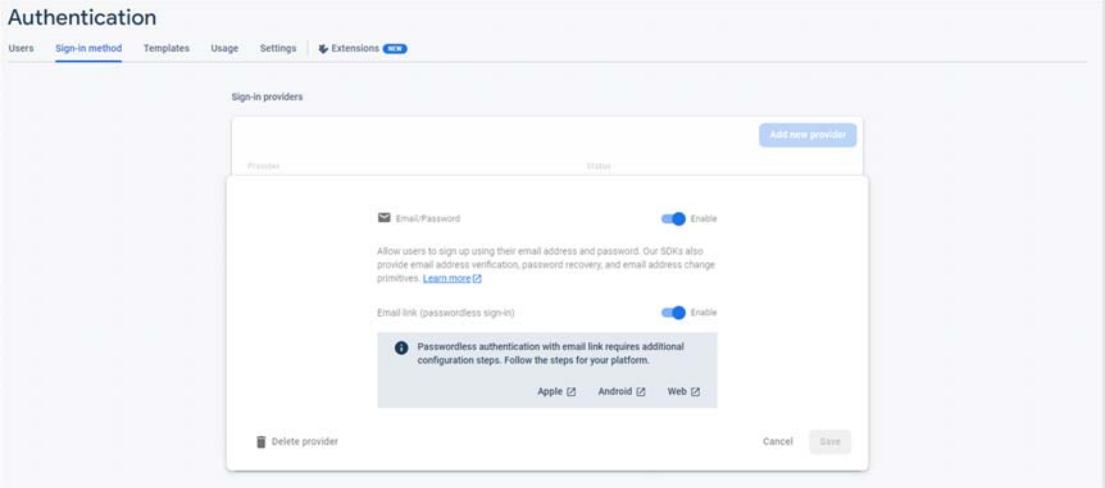


Figure 165. Firebase authentication

To further enhance our security, we conducted a brute-force attack on our system. As one of the most common types of cyberattacks, a successful defense against a brute-force attack serves as a strong indication of our system's security levels.

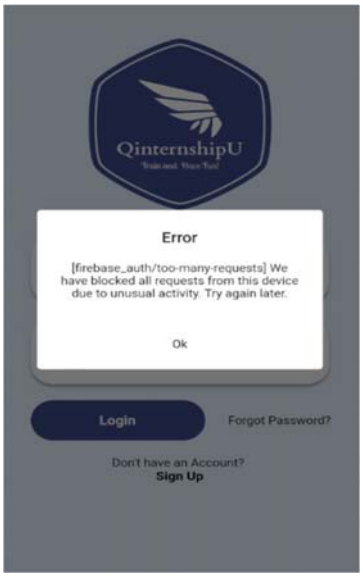


Figure 166. Brute-Force Attack

7.2.6 Availability

Availability is another vital non-functional requirement we addressed in our system. Our tests focused on performing a high volume of read and write actions in a condensed time frame to identify the system's breaking point.

In our tests, the system reached its limit at approximately 50,000 reads in a day. This occurrence, however, was under a scenario of intense and concentrated usage. Moreover, such limitations can be mitigated by upgrading our current plan, thus improving the system's availability under high load.

These tests allowed us to gain a better understanding of our system's limitations and formulate strategies to ensure maximum availability. It is crucial to have these insights to meet user expectations for system availability and maintain a smooth, uninterrupted user experience.

Figure 167. Availability test

7.2.7 Useability

To ensure an optimal user experience, we conducted usability testing of our application. This crucial step in our development process aimed to gauge the ease of use, responsiveness, and visual design of the application from the user's perspective. The feedback gathered would provide insights into areas of improvement and future enhancements.

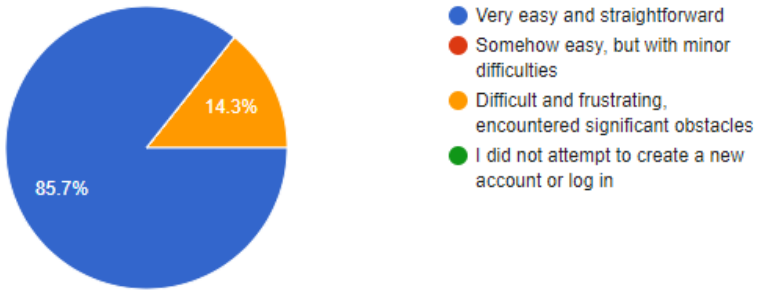


Figure 168. Usability - Registration

Creating and accessing accounts proved to be a seamless process for a substantial majority of users (85.7%), demonstrating user-friendly and straightforward account management. Conversely, a smaller fraction of users (14.3%) faced challenges during this stage, indicating areas that might need refinement to make account creation and logging in more efficient.

Figure 169. Usability - Navigation

Navigating through the application was deemed easy by 42.9% of users, reflecting their ability to discover and utilize desired features with ease. However, 57.1% of users suggested that improvements could be made in the navigation, pointing towards a more intuitive and user-oriented design for future iterations. A few users found the navigation complex or did not explore it thoroughly.

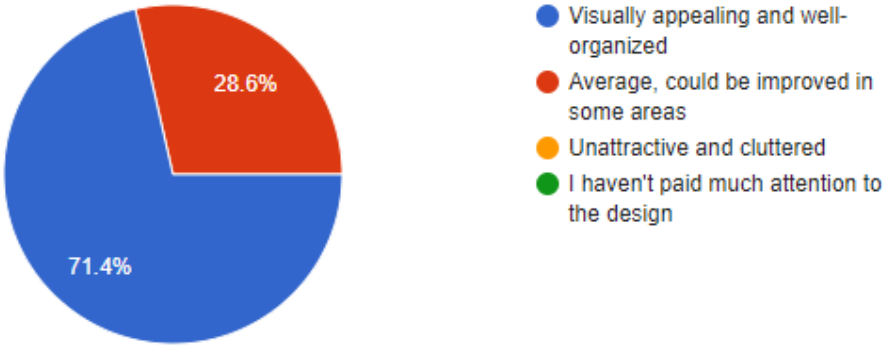


Figure 170. Usability - Visuals

The aesthetic and structural elements of the application were well received, with 71.4% of users appreciating the visual design and organization. This feedback affirms that the design and layout are pleasing and facilitate user comprehension. Nonetheless, 28.6% of users identified room for improvement in the visual design, hinting at potential enhancements in the overall aesthetics.

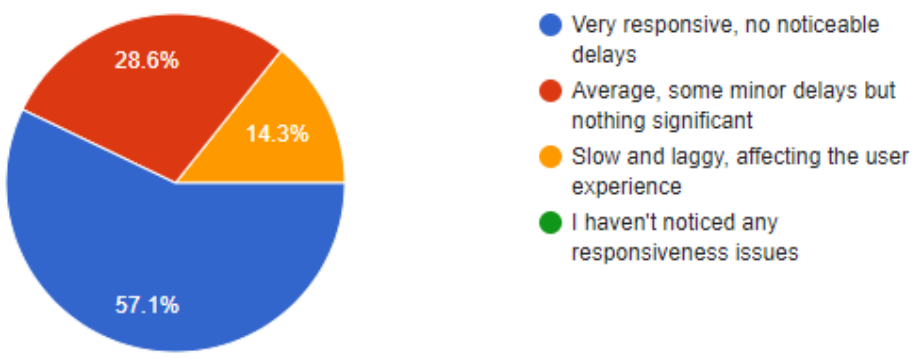


Figure 171. Usability - Responsiveness

The application responsiveness was evaluated positively by 40% of users, who experienced no noticeable delays. Another 40% reported slight delays, and 20% found the application slow, affecting their user experience. These observations underscore the need for performance optimization to promote seamless interactions.

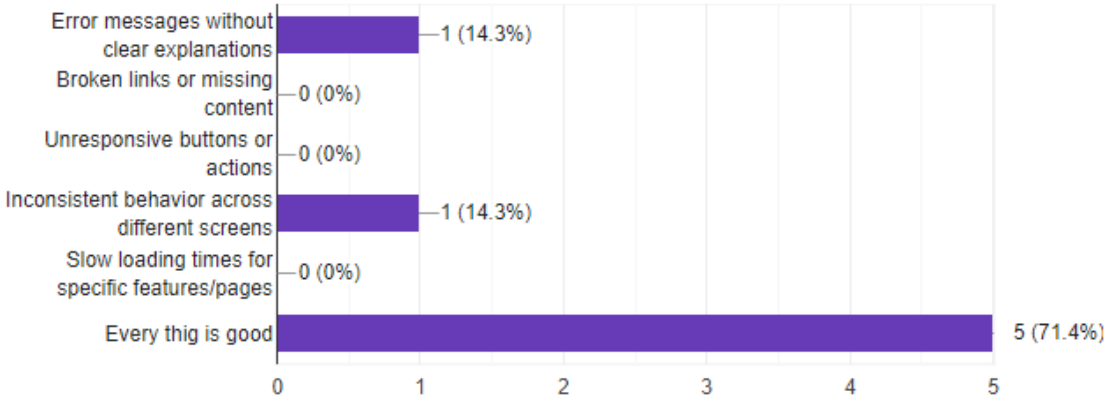


Figure 172. Usability Bugs

During the usability testing, users reported varied issues. 14.3% of users encountered unexplained error messages, and an equal percentage noticed inconsistent behavior across different screens. Specific features or pages were reported to have slow loading times by some users. Despite these issues, a vast majority of users (71.4%) did not face any errors or unexpected behaviors, signaling that the application generally functioned efficiently.

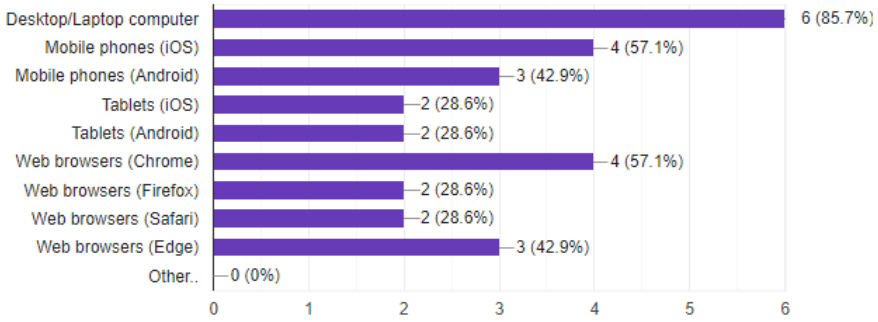


Figure 173. Usability - Compatibility

Usability testing for the application was conducted across a diverse array of devices and platforms. Most users reported optimal compatibility with desktop and laptop computers (85.7%), with a substantial usage on mobile phones as well - 57.1% on iOS and 42.9% on Android. Tablets followed, with both iOS and Android platforms sitting at 28.6%. The application was also subjected to browser compatibility tests on Chrome (57.1%), Firefox (28.6%), Safari (28.6%), and Edge (ranging from 20% to 42.9%). These compatibility assessments spanned a comprehensive set of devices and platforms, aiming to cater to a wide spectrum of users.

Figure 174. Usability - Overall Opinion

As for the overall user-friendliness of the application, the feedback was predominantly positive. Most users (57.1%) gave the application an excellent rating, suggesting that it surpassed their expectations. Another 42.9% considered the application to be good, conceding that there could be areas to refine. A minimal percentage of users were indecisive about the overall usability. This feedback implies a generally favorable user perception of the application's usability, while also pointing towards potential areas that could benefit from further improvements.

8. Conclusion

Our solution offers a streamlined registration process for various user types, an intuitive dashboard to guide users, and user-friendly communication channels between any two users. In addition, we incorporated several value-added features, such as managing requests, creating projects, tracking progress and profile personalization, to enhance the overall user experience.

The strengths of our solution lie in its ease of use, adaptability, and focus on addressing the primary concerns of both coordinators and employers. We were able to identify their needs and expectations, ensuring our solution was closely aligned with their requirements.

However, there were a few shortcomings encountered during the project. As the scope and size of the project expanded, we needed to continually adjust our project plan and priorities. While we successfully adapted to these changes, it highlights the importance of thorough planning and prioritization in future projects.

In conclusion, our project demonstrates the importance of open communication and adaptability in software development. By addressing the primary concerns of users, we were able to deliver a valuable and effective solution that has the potential to significantly improve the internship program experience at Qatar.

9. Future work

We will focus on tuning up the whole user experience and provide versatile functionalities that are not available for now. Although the current user interface is good, there is always room for improvement, so we will work on enhancing the user interface by incorporating more user-friendly design elements such as color-coding, clearer labeling of buttons and tabs, and more intuitive navigation.

We also plan to add a reporting functionality that helps users generate reports based on their internship program data. This can include statistics on intern performance, program effectiveness, and more. This can be achieved by implementing data visualization capabilities such as charts, graphs, and tables to help users better understand their data.

These enhancements will require additional time and planning for the project. However, the benefits of these improvements can be significant in terms of enhancing user experience, increasing program effectiveness, and improving overall satisfaction.

10. Student reflections

Sultan Alemadi:

Throughout my time working on this project, I learned numerous valuable lessons. Initially, the concept of a senior project, which required integrating all our previously acquired knowledge into one comprehensive project, seemed daunting. Questions like "How will I manage my time?", "Will I disappoint my group?", and "How extensive is a senior project?" raced through my mind. However, as time went on, these concerns were gradually addressed. One crucial lesson I learned was that nothing is impossible if others have accomplished it. Whenever I encountered a hurdle, I strove to overcome it, just as others had done before me. Despite my self-doubt, I realized that I might be quite capable.

Throughout this project, I gained many new skills, such as learning a new programming language, creating, and planning diagrams, writing reports, and most importantly, collaborating within a group. These skills will undoubtedly shape my future career. However, I also recognized some key shortcomings. For instance, at times, I noticed that I would lose passion and work inefficiently and slowly. Going forward, I aim to address these weaknesses and continue developing my abilities.

Nawaf Al-Sowadi:

By working on this project, I have learned a lot of things in various aspects. These aspects not only were technical, but also professional. In technical, I have learned a new language that I didn't learn in the university. This new language is very useful if I choose to open a new business for developing applications in various platforms. For example, if the customer asked me to develop an application that works in android, IOS, and web platforms as in most cases in this time, this will be very easy for me. That is because the language I learned in this project will make me write one code only that works in all platforms. In addition, I have learned how to draw diagrams that will illustrate the behavior of the application such as sequence diagram and how to draw diagrams that will illustrate the important use cases in application. These diagrams are very important because they summarize the important things in the application.

In professional aspects, I have learned how to work in a large group (6 students). This could be very useful in my future career since it involves working in a group. It also gives me the experience of leading a group since I was the leader of this project, and this is particularly important because leadership is very useful for anyone seeking advancement in their career. Leadership in this project taught me how to communicate and act with each member differently, as each member has different capabilities. Moreover, I have learned how to be committed to meetings and deadlines and how to speak and act professionally with the supervisors and group members. Also, I have learned how to not copy anything from the internet but instead paraphrase and add the sources because copying is a serious ethical issue.

In this project, there were some drawbacks that I will try not to repeat. For instance, I will evaluate both the project and the team that I will be with because some projects could not be suitable for some project members, or they require effort that team members cannot afford, and some skills are not available to team members. In addition, I will try to be more committed to meetings and

deadlines and avoid miscommunications. Also, I will try to manage my time better and avoid last-minute submissions.

Abdulaziz Al-Kubaisi:

During the course of this project, I have amassed a diverse range of skills and knowledge, spanning both technical and professional domains.

On the technical front, I have gained proficiency in a programming language that was not part of my academic curriculum. This language's utility is immense, especially for entrepreneurial endeavors in the realm of application development across multiple platforms. In the contemporary scenario, clients often request applications compatible with Android, iOS, and web platforms. With the knowledge of this versatile language acquired during the project, I am equipped to code such applications seamlessly. Additionally, I have learned to construct illustrative diagrams like sequence diagrams and use case diagrams that succinctly encapsulate the key facets of an application's functionality.

Turning to professional skills, I have discovered the nuances of collaborating within a sizeable team, in this case, a group of six students. This experience is invaluable considering most career paths necessitate teamwork. I was also the team leader for this project, an opportunity that allowed me to develop vital leadership skills. These skills are indispensable for those seeking progression in their careers. As a leader, I learned to adapt my communication and actions to suit the unique capabilities of each team member. Moreover, the project underscored the importance of adhering to meetings and deadlines, professional decorum, and the avoidance of plagiarism. The latter is crucial as it is a serious ethical transgression.

The project, however, was not devoid of challenges. There were certain shortcomings that I intend to rectify in future endeavors. For instance, I will be more discerning in evaluating both the nature of the project and the abilities of the prospective team members. Some projects might not align with the competencies of certain team members, or they may demand an effort beyond the team's collective capacity. Furthermore, I aim to improve my commitment towards meetings and deadlines, enhance communication, and avoid procrastination. Time management is a skill I intend to refine, particularly to eschew last-minute submissions.

References

[1] Glassdoor. (2013, Nov. 12). *Riipen Overview.* Available: https://www.glassdoor.ca/Overview/Working-at-Riipen-EI_IE1955453.11,17.htm

[2] Rawat. S. (2018, Aug. 17). *What is internshala?* Available: https://www.quora.com/What-is-internshala

[3] Naukrigulf. (2022, Dec. 28). *Naukrigulf.com. Available:* https://www.eyeofriyadh.com/directory/details/6586_naukrigulf-com

[4] TechTarget. (2017). *Model-View-ViewModel (MVVM)*. Available: https://www.techtarget.com/whatis/definition/Model-View-ViewModel

[5] Matt L. (2017). *3 reasons to use the MVVM pattern.* Available: https://www.mrlacey.com/2017/04/3-reasons-to-use-mvvm-pattern.html

[6] Capital. (2019). *The importance of an internship: The top 5 reasons why internships are critical.* Available: https://capital-placement.com/blog/the-importance-of-an-internship-top-5-reasons-why internships-are-critical/#:~:text=An%20internship%20is%20an%20experiential,including%20a%20full%2Dtime%20job

[7] Diane G., Sunita M. (2019). *The Potential Power of Internships and The Impact on Career Preparation.* Available: https://files.eric.ed.gov/fulltext/EJ1263677.pdf

[8] Wilfred V. C. (2017). *The Waterfall Model and the Agile Methodologies: A comparison by project characteristics.* Available: https://www.researchgate.net/publication/313768756_The_Waterfall_Model_and_the_Agile_Methodologies_A_comparison_by_project_characteristics#pf6

[9] NiketKeny. (2019). *A Review on Kotlin and Android Studio Java*. Available: https://www.ripublication.com/ijaerspl2019/ijaerv14n7spl_04.pdf

[10] Aakanksha T., Nisha S., Rushabh G. Trishul G., Pranali C. (2020). *APPLICATION DEVELOPMENT USING FLUTTER*. Available: https://www.irjmets.com/uploadedfiles/paper/volume2/issue_8_august_2020/3180/1628083124.pdf

[11] Claudett M. (2022). *Engineering Design Process.* Available: https://study.com/learn/lesson/engineering-problem-solving-design-process-steps-examples.html

[12] UMBC. (2019, Nov. 28). *What Is an internship?* Available: https://careers.umbc.edu/employers/internships/what-is-an-internship/

[13] Susan M. H. (2022) *What is an Employer?* Available: https://www.thebalancemoney.com/what-is-an-employer-1918113

[14] Matt D. (2020). *How to Find a mentor?* Available: https://www.businessnewsdaily.com/6248-how-to-find-mentor.html

[15] Chegg. (2022. Aug. 2). *Key Roles Involved in Running an Internship Program – Intern Program Tips.* Available: https://www.internships.com/employer/resources/program/keyroles

[16] Glassdoor. (2021. Mar. 3). *Coordinator Job Description*. Available: https://www.glassdoor.com/Job-Descriptions/Coordinator.htm

[17] Education in Qatar Statistical Profile (2016). Gov. Available: https://www.psa.gov.qa/en/statistics/Statistical%20Releases/Social/Education/2016/Education_Statistical_Pro%EF%AC%81le_2016_En.pdf

[18] Owasp Mobile Application Security (2022). *Testing User Privacy Protection*. Available: https://mobile-security.gitbook.io/mobile-security-testing-guide/general-mobile-app-testing-guide/0x04i-testing-user-privacy-protection

[19] Accounting Tools (2022). *Department definition*. Available: https://www.accountingtools.com/articles/department

[20] Eden M. (2022). *Effectiveness of Microsoft Teams: A Study of Perception among Tertiary Learners. Available: https://www.paperpublications.org/upload/book/Effectiveness%20of%20Microsoft-22042022-1.pdf*

Appendix A – Use cases specification

Use case Id: UC**01**	Login
Brief Description	Requires credentials to log into the system.
Primary actors	Trainee, focal point and Admin.
Preconditions: An existing account in the system's database.	
Post-conditions: Granted the user access to the system.	
Main Success Scenario:	
Actor Action	**System Response**
1. Enters his credentials.	2. checks out the database to verify if the account exists. (see 2.a)
	3. checks if the username and password match. (see 3.a)
	4. granting the user access to the system.
Alternative flows: 2.a. If the user does not exist. The system will display that a registration needs to be done. 3.a. If the username and password do not match. An error will appear.	
Special Requirements:	

Use case Id: UC**02**	Registration
Brief Description	Handles the registration of accounts.
Primary actors	Trainee, focal point and Admin.
Preconditions: 5. The user does not have an existing account.	
Post-conditions: 5. Created an account for the user.	
Main Success Scenario:	
Actor Action	**System Response**
1. Click on the Sign-up button.	2. Displays multiple options for different account types.
3. Chooses the suitable account type.	4. Displays the suitable form related to the account type.
5. Fills the form and submit.	6. Checks if the form was fully filled. (see 6.a)
	7. Checks within the database if the email has already been used. (see 7.a.)

	8. Waits for approval from the admin. (8.a.)
	9. Creates the requested account and add it to the database.
	10. Send the credentials to the user via email.
Alternative flows: 6.a. If the form wasn't fully completed, an error window will appear telling the user to complete the form. 7.a. If the email was found. The account will not be created until the user inserts a new unused email. 8.a. If the account wasn't approved. The account won't be registered, and the process will abort.	
Special Requirements:	

Use case Id: UC**03**	Forgot Password
Brief Description	The system will send an email to the user followed by multiple verification steps to set a new password.
Primary actors	Trainee, focal point and Admin.
Preconditions: 5. The user can't login due to a forgotten password.	
Post-conditions: 5. The password of the user had been changed.	
Main Success Scenario:	
Actor Action	**System Response**
1. Clicks on the Forgot password button.	2. Displays a form.
3.Fills the form with correct information.	4. checks if the input by the user was valid and matches what is in the database. (see 4.a)
	5. Sends a link to the user's email to set a new password.
6. Enter the link and set up a new password by typing the new password.	7. Checks if the new password matches the policy. (see 7.a)
	8. Saves the new password in the database.
	9. Sends an email to the user notifying him about the change.
Alternative flows: 4.a. If The user's input was invalid to what is in the database. An error will appear. 7.a. If the password didn't match our policy. An error will appear requesting to follow the password policy.	
Special Requirements:	

Use case Id: UC**04**	Initiate Request
Brief Description	Communication will be established with a selected group of Employers by the coordinator.
Primary actors	coordinator
Preconditions: 5- An existing account in the system's database. 2- The user must be registered as a coordinator.	
Post-conditions: 5. A request will be sent to the selected employers.	
Main Success Scenario:	
Actor Action	**System Response**
1. Clicks on the "send request" button.	2. Displays a list of all the employers in the system.
3. Selects the employers.	
4. Submits the list of employers.	5. Displays a form.
6.Provides the necessary Information and details.	7.Displays the Terms and Conditions agreements and privacy policies.
8. Agrees on the Terms and Conditions agreements and privacy policies.	
9. Confirm and submit the initial request.	10. Sends a request to the selected employers.
Alternative flows:	
Special Requirements:	

Use case Id: UC**05**	Manage Request
Brief Description	Responsible for handling the negotiations of requests between the focal point and the coordinator.
Primary actors	Coordinator, Focal point
Preconditions: 1- The coordinator must be registered as a coordinator to have his privileges. 2- The focal point must be registered as a focal point to have his privileges.	
Post-conditions: 1. The communication between the coordinator and the focal point is terminated and the fulfilled info Is saved in the database.	
Main Success Scenario:	
Actor Action	**System Response**
1. (focal point-coordinator) Navigate notifications	2. Displays the list of the notifications

3. (focal point-coordinator) Choose one of the notifications	4. Display the info about the request with ACCEPT and DENY options
5. (focal point) Chooses to accept (See 5.a)	6. Displays a form
7. (focal point) Provides the necessary information and details and click on submit	8. Displays the Terms and Conditions agreements and privacy policies
9. (focal point) Agrees on the terms and conditions agreements and privacy policies (See 9.a)	10. Save the fulfilled info
	11. Send a notification to the coordinator
12. (coordinator) Repeat steps 1-3	13. Display the focal point's fulfilled form & and a proceed option (See 13.a)
14. (coordinator) Choose the decision he wants	15. Send the more detailed form to the focal point
16. (focal point) Repeat steps 1-3	17. Displays a form
18. (focal point) Provides the necessary information and details and clicks on submit	19. Save the fulfilled info
	20. Send a notification to the coordinator
21. (coordinator) Repeat steps 1-3	22. Display the fulfilled info
Alternative flows: 5.a. if he chooses to deny, the use case is terminated, and the focal point will be navigated to notifications lists and the focal point will receive the decision. 9.a. if he chooses to not agree on the terms and conditions agreements and privacy policies, he will get a popup window says "This process cannot proceed without agreement on the terms and conditions agreements and privacy policies" 13.a. if he chooses to not proceed, the decision will be sent to the focal point as a notification	
Special Requirements: Terms and Conditions agreements and privacy policies	

Use case Id: UC06	Publish Vacancy
Brief Description	Coordinator creates a vacancy for a specific employer.
Primary actors	Coordinator
Preconditions: 1- An existing account in the system's database. 2- The user must be registered as a coordinator.	

3- The data needed for the publishment is available in the database.	
Post-conditions: 1. a new vacancy for the employer was published	
Main Success Scenario:	
Actor Action	**System Response**
1. Click on the “Manage Offers” button from the home page	2. Display two taps: Received Forms and Received Offers
3. Click on Received Offers	4. Display offers from employers (see 4.a)
5. Select the offer	6. Diplay employer and offer information with Publish Vacancy button
7. Click on Publish Vacancy button	8. Display a Confirmation message
9. Click on Yes (see 9.a)	11. Display “Offer submitted to trainees”
	12. Update the offer in database to be displayed in trainees account
Alternative flows: 4.a. if there are no offers, display “No offers arrived yet”. 9.a If click on Cancel, the system will navigate him back.	
Special Requirements:	

Use case Id: UC07	View Offers
Brief Description	Display offers from various employers.
Primary actors	Trainee
Preconditions: 1- The user is logged into the system. 2- The coordinator has published vacancies.	
Post-conditions:	

1. The published offers were displayed to the user.	
Main Success Scenario:	
Actor Action	**System Response**
1. Clicks on the “View offers” button.	2. Extracts the data of the user.
	3. retrieve published offers from the database. (see 3.a)
4. Sends the published offers.	5. Filter the offers according to the user’s data.
	6. Displays the offers.
Alternative flows: 3.a. If there were no offers retrieved from the database. The system will display a message to the user.	
Special Requirements:	

Use case Id: UC08	Apply for Offers
Brief Description	Submits trainee's applications to the employers.
Primary actors	Trainee
Preconditions: 1- The user is logged into the system. 2- The coordinator has published vacancies. 3- The system has filtered the offers based on specific criteria.	
Post-conditions: 1. An application to an offer was filled out and submitted by the Trainee.	
Main Success Scenario:	
Actor Action	**System Response**

1. Select an offer.	2. Displays general information about the offer and an “Apply” button.
3. Clicks on Apply.	4. Displays a form.
5. Fill in the form and submit it.	6. Displays privacy policies, terms and conditions agreements
7. Agrees on the policies and agreements.	8. Submits the user’s application to the selected vacancy. (see 8.a)
	9. Saves in DB
Alternative flows: 8.a. If the user didn’t agree to the displayed terms and conditions. The system will not submit the offer and an error will appear to the user.	
Special Requirements:	

Use case Id: UC09	Assign trainee to projects
Brief Description	Specify a project for a trainee.
Primary actors	Mentor
Preconditions: 1- The user is logged into the system. 2- There are accepted applicant/s in the offer. 3- There are available projects to sign in.	
Post-conditions: 1. The trainee is signed into a project successfully.	
Main Success Scenario:	
Actor Action	**System Response**
1. Navigate to projects.	2. Display a list of pre-created projects.

3. Clicks on one of the projects.	4. Displays information about the project with the option to add and change trainees.
5. Click add trainee. (See 5.a)	6. Display a list of accepted trainees
7. Click on the wanted to assign trainee and click Add button.	8. Send a notification to the assigned trainees.
	9. Save the info in DB
Alternative flows: 5.a. If the mentor clicks on change, the system will show a list of the assigned trainee to change.	
Special Requirements:	

Use case Id: UC10	Create project
Brief Description	Mentor creates a project for the trainee to be assigned.
Primary actors	Mentor
Preconditions: 1- The user is logged into the system. 2- Mentor is assigned to offer.	
Post-conditions: 1. A project is created successfully.	
Main Success Scenario:	
Actor Action	**System Response**
1. Navigate to projects.	2. Display a list of pre-created projects, if any, with the option to create a new project.
3. Clicks on creating a new project.	4. Displays a form to fill in with the info about the project.
5. Fill in the necessary info and click on Save.	6. Display a pop-up window for confirming the successful process

	7. Save the info in DB.
Alternative flows:	
Special Requirements:	

Use case Id: UC11	View project
Brief Description	Trainee can see his assigned projects.
Primary actors	Trainee
Preconditions: 1- The user is logged into the system. 2- There are available projects to view.	
Post-conditions: 1. Trainee saw his projects	
Main Success Scenario:	

Actor Action	**System Response**
1. Navigate to projects.	2. Display a list of assigned projects. (See 2.a)
3. Clicks on one of the assigned projects.	4. Display a dashboard for the project with a menu bar to submit the assignments and track progress.

Alternative flows: 2.a. if the trainee is not assigned to any projects, a “Sorry, you are not having any assigned project” message will appear.
Special Requirements:

Use case Id: UC12	Complete tasks
Brief Description	Trainee can complete his tasks in his projects.
Primary actors	Trainee
Preconditions: 1- The user is logged into the system.	

2- The trainee assigned to a project. 3- The trainee has tasks to complete.	
Post-conditions: 1. Trainee complete his tasks in projects.	
Main Success Scenario:	
Actor Action	**System Response**
1. Navigate to home page.	2. Display the project tasks and assignments. (See 2.a)
3. Clicks on one of the tasks.	4. Display confirmation message.
5. Click yes.	6. Change the status of the task to “Done”
	7. Save the new status in the database
Alternative flows: 2.a. if the trainee is not assigned to any project, a “Sorry, you are not assigned to any project yet” message will appear.	
Special Requirements:	

Use case Id: UC13	Submit trainee reports
Brief Description	Trainee submits his reports to the system.
Primary actors	Trainee
Preconditions: 1- The user is logged into the system. 2- There is an available project to submit the report for. 3- The trainee did the report and has a file for it.	
Post-conditions: 1. Trainee submitted his report on his assigned projects.	
Main Success Scenario:	

Actor Action	System Response
1. Navigate to projects.	2. Display a list of assigned projects. (See 2.a)
3. Clicks on one of the assigned projects.	4. Display a dashboard for the project with a menu bar with Assignments and Track Progress options.
5. Hover over the Assignments.	6. Display “Reports” and “Feedbacks”.
7. Click on Reports.	8. Display a list of reports to submit. (See 8.b)
9. Click on one of the reports.	10. Navigate to the upload page.
11. Upload the report file and click on submit.	12. Display success message of the submitted report. (See 12.c)
	13. Save the report in DB.

Alternative flows:

2.a. if the trainee is not assigned to any projects, a “Sorry, you are not having any assigned project” message will appear.

8.b. if there are no reports to submit, a “You do not have any reports to submit!” message will appear.

12.c. if there is a problem with the uploading, a message asking for trying again will appear.

Special Requirements:

Use case Id: UC14	Submit trainee feedback
Brief Description	Trainee submits his feedback to the system.
Primary actors	Trainee

Preconditions:

1- The user is logged into the system.

2- There is an available project to submit feedback on.

3- The trainee has the info to fill in the form of the feedback with.

Post-conditions:

1. Trainee submitted feedback on his assigned projects.

Main Success Scenario:

Actor Action	System Response
1. Navigate to projects.	2. Display a list of assigned projects. (See 2.a)
3. Clicks on one of the assigned projects.	4. Display a dashboard for the project with a menu bar with Assignments and Track Progress options.
5. Hover over the Assignments.	6. Display “Reports” and “Feedbacks”
7. Click on Feedbacks	8. Display a list of feedbacks to submit. (See 8.b)
9. Click on one of the types of feedback.	10. Navigate to feedback form.
11. Fill in the necessary info and click on submit.	12. Display success message of the submitted report.
	13. Save the feedback in DB.
Alternative flows: 2.a. if the trainee is not assigned to any projects, a “Sorry, you are not having any assigned project” message will appear. 8.b. if there are no feedbacks to submit, a “You do not have any feedbacks to submit!” message will appear.	
Special Requirements:	

Use case Id: UC15	Submit evaluation
Brief Description	Mentors submits his evaluation about the trainee.
Primary actors	Trainee
Preconditions: 1- The user is logged into the system. 2- The mentor is responsible for a trainee/s. 3- The mentor has the info to fill in the form of the evaluation with.	
Post-conditions: 1. Mentor submitted the evaluation on his assigned trainee/s.	
Main Success Scenario:	

Actor Action	System Response
1. Navigate to Evaluations.	2. Display a list of assigned trainees. (See 2.a)
3. Clicks on one of the trainees.	4. Display info about the trainee with the option to submit an evaluation.
7. Click on Submit Evaluation	8. Display a list of evaluations to submit. (See 8.b)
9. Click on one of the evaluations.	10. Navigate to evaluation form.
11. Fill in the necessary info and click on submit.	12. Display success message of the submitted evaluation.
	13. Save the evaluation in DB.
	14. Send a notification to the trainee about the evaluation.
Alternative flows: 2.a. if the mentor has no trainee/s, a “Sorry, you do not have any assigned trainee/s” message will appear. 8.b. if there are no evaluations to submit, a “You do not have any evaluations to submit!” message will appear.	
Special Requirements:	

Use case Id: UC16	Manage Application
Brief Description	Manage the state of the trainee’s application to a specific offer.
Primary actors	Focal Point, Trainee
Preconditions: 1- The user is logged into the system. 2- An application was already made to the offer by a trainee.	
Post-conditions: 1. A decision was made by the Focal point regarding the submitted application.	
Main Success Scenario:	

Actor Action	System Response
	1. Displays a list of applications made for the offer.
2. Selects an application.	3. Displays extra information about the application.
4. Request access for the private profile of the applicant.	5. Send a notification to the coordinator about the request.
6. Trainee- Accepts the request	7. Access to the private profile will be granted to the focal point and he will be notified. (see 7.a)
8. Accepts the user's application.	9. Assign the trainee to the vacancy. (see 9.a)
Alternative flows: 7.a. If the trainee denied the focal point's request. Access won't be granted to the selected private profile. 9.a. If the focal point denied the application. The trainee won't be assigned to a vacancy and a notification will be sent regarding the decision.	
Special Requirements:	

Use case Id: UC17	View Grades
Brief Description	Displays a trainee's grade to the coordinator.
Primary actors	Coordinator
Preconditions: 1- The user is logged into the system. 2- The user has students under his coordination.	
Post-conditions: 1. Grades of the selected trainee will be viewed.	
Main Success Scenario:	

Actor Action	System Response
1.Navigate to "View Grades"	2. Displays a list of trainees. (see 2.a)
3. Selects a trainee.	4.Displays the grade of the selected trainee.

Alternative flows: 2.a. If the coordinator does not have any trainees under their coordination. The system will display a notice.
Special Requirements:

Use case Id: UC18	Establish Communication
Brief Description	Communication between two parties will be established.
Primary actors	Coordinator, Trainee, Employer and Examiner
Preconditions: 1- The user is logged into the system.	
Post-conditions: 1. Contact between two different users will be made.	
Main Success Scenario:	
Actor Action	**System Response**
1. Navigate to “Contact”.	2. Extracts the user’s data.
	3. Displays a list of users who are associated with the current user. (see 3.a)
4. Selects a user.	5. Displays a text field with minor details about the selected user.
6. Types out the message in the text field.	
7. Submits the message.	8. Sends a notification to the selected user.
	9. Establish a connection between the two users.
Alternative flows: 3.a. If there are no associated users. A notice will appear, and nothing will be displayed.	
Special Requirements:	

Use case Id: UC19	Select Examiner

Brief Description	The coordinator will be able to select an examiner and assign students to the examiner.
Primary actors	Coordinator
Preconditions: 1. The user is logged into the system. 2. Examiners and trainees are available	
Post-conditions: 1. An examiner was selected, and students were assigned to them.	
Main Success Scenario:	

Actor Action	**System Response**
1. Navigate to “Select examiner”.	2. Displays a list of available examiners. (see 2.a)
3. Selects an examiner.	4. Displays a list of students who do not have an examiner. (see 4.a)
5. Selects trainees and confirm.	6. Assign the selected students to the examiner.
	7. Saves to the database.
	8. Sends a notification to the trainees and examiners.

Alternative flows: 2.a. If there are no available examiners. A notice will appear, and the page will be empty. 4.a. if all students have an examiner. A notice will appear, and the page will be empty.
Special Requirements:

Use case Id: UC20	Submit Grades
Brief Description	A trainee's grade will be submitted by the examiner.
Primary actors	Examiner
Preconditions: 1- The user is logged into the system.	

Post-conditions: 1. A grade will be uploaded and submitted by the examiner.	
Main Success Scenario:	
Actor Action	**System Response**
1. Navigate to “Submit Grades”.	2. Extracts the user’s data.
	3. Displays a list of trainees who are associated with the current user. (see 3.a)
4. Selects a trainee.	5. Displays a form.
6. Fill out the form with the grade and required information.	
7. Submits the form.	8. Sends a notification to the coordinator.
	9. Saves the form in the database.
Alternative flows: 3.a. If there are no associated trainees. A notice will appear, and nothing will be displayed.	
Special Requirements:	

Use case Id: UC21	Fill site visit
Brief Description	The coordinator will fill out the site visit.
Primary actors	Coordinator
Preconditions: 1- The user is logged into the system.	
Post-conditions: The site visit was filled in by the coordinator and was saved in the database.	
Main Success Scenario:	
Actor Action	**System Response**

1. Fills in the site visit form.	2. Verify the user's input
	3. Saves the form to the database. (see 3.a)
	4. Displays a completion notice.
Alternative flows: 3.a. If the form was not fully filled. An error will appear.	

Use case Id: UC22	Manage Profile
Brief Description	Display the profile Information with the ability to customize the profile.
Primary actors	Employer, Focal Point, Trainee, Mentor and Coordinator
Preconditions: 1. The user is logged into the system.	
Post-conditions: User's information was displayed with options to customize them.	
Main Success Scenario:	

Actor Action	**System Response**
1. Navigate to "Manage profile"	2. Displays user's information and profile customization types and options. (see 2.a)
3. Selects an option.	4. Display a dialog to change the information
5. Enter necessary information	6. Update information in database (see 6.a)
Alternative flows: 2.a. If the user was a trainee. The system displays a private profile customization option. 6.a. Error message will appear if the user entered incorrect information.	

Use case Id: UC23	Customize private profile
Brief Description	The user will be able to customize his private profile.
Primary actors	Trainee
Preconditions: 1.The user is logged in to the system.	

2.The account type must be a trainee.	
Post-conditions: The user's private profile was updated.	
Main Success Scenario:	
Actor Action	**System Response**
1.Clicks on "Edit private Profile".	2. Displays the private profile of the user with edit options.
3. Hover over the edit icon and click on it.	4. Enable the user to edit his selection.
5. Change and edit the information.	
6. Click on save.	7. Verify the user's input.
	8. Update the information in the database. (see 7.a)
	9. Refreshes the page.
Alternative flows: 8.a. If the new information does not match the regulations and rules. The system will display an error.	

Use case Id: UC24	Customize public profile
Brief Description	The user will be able to customize his public profile.
Primary actors	Trainee, Employer and Mentor
Preconditions: 1.The user is logged in to the system	
Post-conditions: The Public profile was customized by the user.	
Main Success Scenario:	
Actor Action	**System Response**
1.Clicks on "Edit Public Profile".	2. Displays the public profile of the user with edit options.
3. Hover over the edit icon and click on it.	4. Enable the user to edit his selection.

5. Change and edit the information.	
6. Click on save.	7. Verify the user's input.
	8. Update the information in the database. (see 8.a)
	9. Refreshes the page.
Alternative flows: 8.a. If the new information does not match the regulations and rules. The system will display an error.	

Use case Id: **UC25**	Change Password
Brief Description	change the password of the account.
Primary actors	Trainee, focal point, employer and mentor.
Preconditions: 1. The user must have an active account. 2. Know the username and old password	
Post-conditions: The password was changed successfully	
Main Success Scenario:	

Actor Action	**System Response**
1.Clicks on "Change Password"	2.Displays a security question.
3.Fill the security question	4.Validate the security question. (See 3.a)
	5.Asks the user to enter his old and new password and confirm it.
6.Enters the old and new password and confirm it	7. Validate the passwords (See 7.a) (See 7.b) (See 7.c)
	8.Save the new password to the Database
	9.Send a notification that the password has been changed successfully
Alternative flows: 3.a. if the answer is not correct, the system will display mismatch message	

7.a. If the new password is the same as the old password, ask the user to enter another password. 7.b. if the new password does not meet the requirements for a strong password, a rechose another password message will be displayed. 7.c. if the new password is different in confirmation, error message will be displayed.

Use case Id: UC26	Manage account
Brief Description	Display the account menu of selections that the user can enter any of the options like manage profile, change settings and view application status.
Primary actors	Employer, Focal Point, Trainee, Mentor and Coordinator
Preconditions:	
Post-conditions: A menu displays to the user and can select one of the options	
Main Success Scenario:	

Actor Action	**System Response**
1. User logs in. <include Login use case>	
2. User navigates to “Manage account”	3. Displays the account menu
4. User chooses one of the options in the menu	5. Displays the selected option
Alternative flows:	

Use case Id: UC27	View application status
Brief Description	Trainee can view the application status of the projects he applied for.
Primary actors	Trainee
Preconditions: 1- The trainee is logged into the system. 2- The trainee has already applied for an offer	
Post-conditions:	
Main Success Scenario:	

Actor Action	System Response
1. Navigate to “View application status”	2. Display the status of the offers the trainee applied to. If the trainee did not apply for an offer before, then (See 2.a)
3. Trainee can view all his applications status with details	

Alternative flows:

2.a. The system shows this message “No applications to shown, you should apply for an offer”. <include Apply for offer/s use case>

Use case Id: UC28	View submitted reports
Brief Description	This use case is for the coordinator to view the submitted reports by the trainees.
Primary actors	Coordinator

Preconditions:

1- The user is logged into the system.

2- Trainee submitted the report.

Post-conditions:

Main Success Scenario:

Actor Action	System Response
1. Navigate to trainees	2. Display a list of trainees
3. Chose trainee	4. Display a list of options
5. Chose report type	6. Open file viewer (see 5.a)

Alternative flows:

6.a. if the report is not submitted, a message saying “Trainee have not filled the form” will appear.

Use case Id: UC29	View final report
Brief Description	The examiner will view the final report that was submitted by the trainees.
Primary actors	Examiner
Preconditions: 1- The user is logged into the system.	
Post-conditions: 1- Final report is reviewed by the examiner	
Main Success Scenario:	
Actor Action	**System Response**
1. Navigate to “View Final Report”.	2. List of the submitted final reports by the trainees will display for the examiner
3. Selects a trainee to view his final report	4. Displays the final report of the trainee
5. Examiner reviews and grades the final report	6. Final report is graded and saved in the database
	7. Sends a notification to the coordinator.
Alternative flows:	

Use case Id: UC30	View presentation
Brief Description	The examiner will view the presentation submitted by the trainees.
Primary actors	Examiner
Preconditions: 1- The user is logged into the system.	
Post-conditions: 1- Trainee presentation is reviewed by the examiner	
Main Success Scenario:	
Actor Action	**System Response**
1. Navigate to “View presentation”.	2. List of the submitted presentations by the trainees will display for the examiner

3. Selects a trainee to view his presentation	4. Displays the presentation of the trainee
5. Examiner reviews and grades the presentation	6. The presentation is graded and saved in the database
	7. Sends a notification to the coordinator.
Alternative flows:	

Use case Id: UC31	View evaluation rubric
Brief Description	Trainees will view the evaluation rubric to give feedback about the mentor and t The examiner and mentor will view the evaluation rubric to evaluate the trainee.
Primary actors	Examiner & Mentor
Preconditions: 1. The user is logged into the system. 2. Examiner and mentor have trainee to evaluate.	
Post-conditions: An evaluation rubric will be displayed to the mentor and examiner to evaluate the trainee.	
Main Success Scenario:	
Actor Action	**System Response**
1. Navigate to evaluate trainee	2. Displays the evaluation rubric
3. Evaluate the trainee and click on submit	4. Display “Evaluation submitted successfully”
Alternative flows:	

Appendix B – Test cases specification

Test Case Template (Example)

Test case #	3.1	**Associated user case ID**	U3
Test designed by	Abbas Ibn Firnas	**Test design date**	15/01/2023
Executed by	Abbas Ibn Firnas	**Execution date**	20/02/2023
Test case name	ATM change PIN		
Short description	Test the ATM change PIN use case		

Pre-conditions:
- The user has a valid ATM card
- The user has accessed the ATM by placing his ATM card in the machine
- The current PIN is 1234
- The system displays the main menu

Step	Test Step	Expected System Response	Actual Result	Pass/ Fail	Comment
1	Click the 'Change PIN' button	The system displays a message asking the user to enter the new PIN	As expected	Pass	
2	Enter '5555'	The system displays a message asking the user to confirm (re-enter) the new PIN	As expected	Pass	
3	Re-enter '5555'	The system displays a message of successful operation The system asks the user if he wants to perform other operations	As expected	Pass	
4	Click 'YES' button	The system displays the main menu	As expected	Pass	
5	Check post-condition 1				
6	Repeat steps 1,2,3 using another PIN say '6666' and click 'NO' button	The system is exited and displays a greeting message asking the user to place his ATM card in the machine			
7	Check post-condition 2				
8	Repeat steps 1,2, using another PIN say '7777'	The system displays a message asking the user to confirm (re-enter) the new PIN			
9	Enter a wrong confirmation (say '9876')	The system displays a message of unsuccessful operation and asks the user to confirm the correct PIN			
10	Re-enter '7777'	The system displays a message of successful operation The system asks the user if he wants to perform other operations			
11	Click 'NO' button	The system is exited and displays a greeting message			
12	Check post-condition 3				

Post-conditions:
1. The new PIN '5555' is saved in the database
2. The new PIN '6666' is saved in the database
3. The new PIN '7777' is saved in the database

Test case #	3.1	**Associated user case ID**	U1
Test designed by	Nawaf Al-Sowadi	**Test design date**	12/05/2023
Executed by	Nawaf Al-Sowadi	**Execution date**	18/05/2023
Test case name	Login		
Short description	Test the login use case.		

Pre-conditions:
- The user has an account to login to.

Step	Test Step	Expected System Response	Actual Result	Pass/ Fail	Comment
1	Enter email “john.doe” and password “password” and click login	Display error message saying “Incorrect Email and Password”	As expected	Pass	
2	Click Ok button	Pop back to login page	As expected	Pass	
3	Repeat steps 1, 2 with email “nonexistinguser@example.com” and password “wrongpassword”	Display error message saying “There is no user record corresponding to this identifier. The user may have been deleted.”	As expected	Pass	
4	Repeat steps 1, 2 with email “ada@qu.edu.qa” and password “123456”	Display trainee home page	As expected	Pass	
5	Check post-condition 1				
6	Repeat steps 1, 2 with email “ntest@qu.edu.qa” and password “123456”	Display coordinator home page	As expected	pass	
7	Check post-condition 2				
8	Repeat steps 1, 2 with email “qu.eduqa@qu.edu.qa” and password “123456”	Display employer home page	As expected	pass	
9	Check post-condition 3				
10	Repeat steps 1, 2 with email “menthor@hotmail.com” and password “699282”	Display coordinator home page	As expected	pass	
11	Check post-condition 4				
12	Repeat steps 1, 2 with email “admin@qu.edu.qa” and password “admin123”	Display admin home page	As expected	pass	
13	Check post-condition 5				
14	Repeat steps 1, 2 with email “Exo@qu.edu.qa” and password “123456”	Display examiner home page	As expected	pass	
15	Check post-condition 6				

Post-conditions:
1- Trainee granted access to his account.

2- Coordinator granted access to his account.
3- Employer granted access to his account.
4- Mentor granted access to his account.
5- Admin granted access to his account.
6- Examiner granted access to his account.

Test case #	3.2	**Associated user case ID**	U2
Test designed by	Nawaf Al-Sowadi	**Test design date**	10/05/2023
Executed by	Nawaf Al-Sowadi	**Execution date**	18/05/2023
Test case name	Registration		
Short description	Test the registration use case		

Pre-conditions:
- The register knows his role in the internship
- The Trainee department has already registered before the trainee.

Step	**Test Step**	**Expected System Response**	**Actual Result**	**Pass/ Fail**	**Comment**
1	Click on Sign Up button	Navigate to sign up page and display 3 choices of registration: Trainee, Employer, and Department.	As expected	Pass	
	Click on Continue button without selection type of registration	Display error message saying "Please select your role before continue"	As expected	Pass	
3	Chose Trainee/Employer/Department	Display Email, password, and Confirm Password fields	As expected	Pass	
4	Enter badly formatted Email, Password, and Confirm Password and click on Continue	Display error message saying "Email address is badly formatted"	As expected	Pass	
5	Enter existed Email, Password, and Confirm Password and click on Continue	Display error message saying "The email address is already in use by another account"	As expected	Pass	
6	Enter Email, weak Password, and Confirm Password and click on Continue	Display error message saying "The password must contain at least 6 characters"	As expected	Pass	
7	Enter Email, unmatched Password, and Confirm Password and click on Continue	Display error message saying "Passwords do not match"	As expected	pass	
8	Enter Email, Password, and Confirm Password and click on Continue	Display setup account information to fill	As expected	pass	
9	Check post-condition 1				
10	Fill but not all necessary information and click on Submit	Display error message saying "Fill empty fields!"	As expected	pass	
11	Fill all necessary information and click on Submit	(If trainee) Display "User was successfully created! Verify your email before login"	As expected	pass	While the employer registration

		(If employer/department) Display "User was successfully created!"			functionality itself is functioning correctly, we have encountered an issue that is preventing the test from passing. Through thorough investigation and error searching, we have determined that the problem lies with the library and test widget we are currently utilizing.
12	Check post-condition 2				
Post-conditions: **1- Account created successfully.** **2- Finish setting up account.**					

Test case #	3.3	**Associated user case ID**	U16
Test designed by	Nawaf Al-Sowadi	**Test design date**	12/05/2023
Executed by	Nawaf Al-Sowadi	**Execution date**	18/05/2023
Test case name	Manage Application		
Short description	Test the registration use case.		

Pre-conditions:
- The employer logged in to his account.
- There is an offer from an employer for trainees to apply for.
- There are applicants to employer offer.

Step	Test Step	Expected System Response	Actual Result	Pass/ Fail	Comment
1	Click on a specific offer	Display two taps: Offer Information and Offer Applications	As expected	Pass	
2	Click on Offer Applications	Display a list of applicants with their application status	As expected	Pass	
3	Click on applicant	Display public information for the trainee with request access option to view private information	As expected	Pass	
4	Click on request access	Send trainee notification about request access	As expected	Pass	

5		Display “A request has been sent to the Trainee to have access on his private information.\nYou will be notified as soon as possible.”	As expected	Pass	
6	Repeat step 4				
7	Click Ok button	Pop up back	As expected	Pass	
8	Click Back arrow button on app bar	Pop up back	As expected	pass	
9	Click on Accept button	Display “Are you sure you want to accept this trainee?”	As expected	pass	
10	Click on Yes	Change the status of the applicant to Accepted	As expected	pass	
11	Check post-condition 2				
12	Repeat step 8	Display “This trainee already accepted”	As expected	pass	
13	Click on Reject button	Display “Are you sure you want to reject this trainee?”	As expected	pass	
14	Click on Yes	Change the status of the applicant to Rejected	As expected	pass	
15	Check post-condition 2				
16	Repeat step 14	Display “This trainee already rejected”	As expected	pass	

Post-conditions:

1- **Update the status of trainee to Accepted, notify the trainee, and make the trainee available to be signed for a project by the mentor.**
2- **Update the status of trainee to Rejected and notify the trainee.s**

Test case #	3.4	**Associated user case ID**	U13
Test designed by	Sultan Alemadi	**Test design date**	18/05/2023
Executed by	Sultan Alemadi	**Execution date**	18/05/2023
Test case name	Submit Trainee Reports		
Short description	Trainee submits his reports to the system.		

Pre-conditions:
- The trainee is logged in.
- The trainee is assigned into a project.
- The coordinator opened the submission.

Step	Test Step	Expected System Response	Actual Result	Pass/ Fail	Comment
1	Click the on the report card	The system displays a pop-up dialog that asks, “Do you want to submit a file?”.	As expected	Pass	
2	Clicks on “Upload Files”	The system initiates the native file picker dialog (Windows Explorer for PC, Finder for macOS, or File Picker for Android/iOS) on the user's device.	As expected	Pass	Only the suitable file format will be displayed for submission. Other will be hidden from the users.
3	Picks a file and confirm	The system displays a pop-up dialog that confirming the submission “File has been uploaded successfully!”	As expected	Pass	
4	Click “Ok” button	The system displays the projects screen.	As expected	Pass	
5	Click the on the report card	The system displays a pop-up dialog that asks, “You have already uploaded a file. Do you want to reupload?”.	As expected	Pass	
6	Clicks “Yes”	The system displays a pop-up dialog that asks, “Do you want to submit a file?”.	As expected	Pass	
7	Clicks on “Upload Files”	The system initiates the native file picker dialog (Windows Explorer for PC, Finder for macOS, or File Picker for Android/iOS) on the user's device.	As expected	Pass	
8	Picks a file and confirm	The system displays a pop-up dialog that confirming the submission “File has been uploaded successfully!”	As expected	Pass	

Post-conditions:
1. The file is uploaded to the firebase storage.

Test case #	3.5	Associated user case ID	U14
Test designed by	Sultan Alemadi	**Test design date**	18/05/2023
Executed by	Sultan Alemadi	**Execution date**	18/05/2023
Test case name	Submit trainee feedback		
Short description	The trainee submits his feedback to the system.		

Pre-conditions:
- The trainee is logged in.
- The trainee is assigned into a project.
- The coordinator opened the feedback submission.

Step	Test Step	Expected System Response	Actual Result	Pass/ Fail	Comment
1	Click the on the feedback card	The system navigates the user to the feedback screen.	As expected	Pass	
2	Clicks on “Submit”	The system displays a red text on each of the 6 cards “Please enter your Information”.	As expected	Pass	
3	Enters suitable information in one field			Pass	
4	Clicks on “Submit”	The system displays a red text on the 5 cards “Please enter your Information”.	As expected	Pass	
5	Enters suitable information in all fields				
6	Clicks on “Submit”	The system displays a snack-bar that says, “Feedback has been uploaded successfully	As expected	Pass	
7	Click the on the feedback card	The system displays a pop-up dialog that asks, “You have already submitted your feedback evaluation.”	As expected	Pass	
8	Clicks on “Ok”	The system displays the projects page.	As expected	Pass	

Post-conditions:
1. The file is uploaded to the firebase database.
2. The trainee can not access the feedback.

Test case #	3.6	Associated user case ID	UC22
Test designed by	Abdulaziz Al-kubaisi	**Test design date**	10/05/2023
Executed by	Abdulaziz Al-kubaisi	**Execution date**	18/05/2023
Test case name	Manage Profile		
Short description	Test to change profile phone number		

Pre-conditions:

- The user is logged into the system.

Step	Test Step	Expected System Response	Actual Result	Pass/ Fail	Comment
1	Navigate to “Profile page”	Display the profile	As expected	Pass	
2	Press the customize option change the phone number	Display a pop-up window to enter the new phone number	As expected	Pass	
3	Type the phone number		As expected	Pass	
4	Press the bottom to save the new update	Save the new data in the database and update the profile	As expected	Pass	

Post-conditions:

1. phone number updated.

Test case #	3.7	**Associated user case ID**	UC**03**
Test designed by	Abdulaziz Al-kubaisi	**Test design date**	10/05/2023
Executed by	Abdulaziz Al-kubaisi	**Execution date**	18/05/2023
Test case name	Forgot Password		
Short description	The system will send an email to the user followed by multiple verification steps to set a new password.		

Pre-conditions:

- ❖ The user can’t login due to a forgotten password.

Step	Test Step	Expected System Response	Actual Result	Pass/ Fail	Comment
1	Clicks on the Forgot password button	Display a form	As expected	Pass	

2	Filles the form with correct information	checks if the input by the user was valid and matches what is in the database	As expected	Pass	
3		Sends a link to the user's email to set a new password.	As expected	Pass	
4	Enter the link and set up a new password by typing the new password.	Checks if the new password matches the policy.	As expected	Pass	
		Saves the new password in the database.	As expected	Pass	
		Sends an email to the user notifying him about the change.	As expected	Pass	
Post-conditions: **1. password has been changing successfully.**					

Test case #	3.8	**Associated user case ID**	UC**20**
Test designed by	Abdullah Ahmed	**Test design date**	15/05/2023
Executed by	Abdullah Ahmed	**Execution date**	15/05/2023
Test case name	Establish communication		
Short description	The user log-in to his account and start communicating		

Pre-conditions:

- ❖ The user has account in the system
- ❖ The user is logged into the system.

Step	**Test Step**	**Expected System Response**	**Actual Result**	**Pass/ Fail**	**Comment**
1	Enter the email and password	The system checks the database for the account and let the user in to the home page.	As expected	Pass	

2	Press the chatting icon	Display the chatting home page.	As expected	Pass	
3	Press the search icon to search for other User to contact.	Display a list of users	As expected	Pass	
4	Click on the specified user to contact	Display the chatting page and the chat history if there is	As expected	Pass	
	Enter a massage “Hi “				
	Click the send bottom	Send the massage and display it for the second user	As expected	Pass	
Post-conditions: **1. communication have been done between two users**					

Test case #	3.9	**Associated user case ID**	UC**20**
Test designed by	Abdullah Ahmed	**Test design date**	15/05/2023
Executed by	Abdullah Ahmed	**Execution date**	15/05/2023
Test case name	Apply for offer		
Short description	The user log-in to his account and choose the best offer for him to apply		

Pre-conditions:

- ❖ The user has account in the system
- ❖ The user is logged into the system.

Step	**Test Step**	**Expected System Response**	**Actual Result**	**Pass/ Fail**	**Comment**
1	Enter the email and password	The system checks the database for the account and	As expected	Pass	

		let the user in to the home page.			
2	Press the offer icon in the navigation bar.	Display the chatting home page.	As expected	Pass	
3	Choose an offer among offers	Display offer details.	As expected	Pass	
4	Press the apply for offer bottom	Display the pop-up window for the user to write the statement in the text field	As expected	Pass	
	Write the statement and press apply	Show massage "you have been successfully applied for the offer"			
		Save the offer in the database and send the user application to the employer	As expected	Pass	

Post-conditions:

1. apply for the offer

employer received the application from that user

Test case #	4	Associated user case ID	UC25
Test designed by	Mohammed Al-mohannadi	Test design date	18/05/2023
Executed by	Mohammed Al-mohannadi	Execution date	18/05/2023
Test case name	Fill site visit		
Short description	The coordinator will fill out the site visit.		

Pre-conditions:
- The user chose this form type from the manage forms page.
- The user is logged into the system.

Step	Test Step	Expected System Response	Actual Result	Pass/ Fail	Comment
1	Enter the email and password	The system checks the database for the account and let the user in to the home page.	As expected	Pass	
2	chose visit form from the manage forms page	Display the visit site form	As expected	Pass	

3	User fill the form	Verify the user's input and Display a complete message	As expected	Pass	

Post-conditions:
1. The site visit was filled in by the coordinator and was saved in the database.

Test case #	4.1	Associated user case ID	UC10
Test designed by	Mohammed Al-mohannadi	Test design date	18/05/2023
Executed by	Mohammed Al-mohannadi	Execution date	18/05/2023
Test case name	Create project		
Short description	Focal point creates a project for the trainee to be assigned.		

Pre-conditions:
- There are available projects to sign in.
- The user is logged into the system.

Step	Test Step	Expected System Response	Actual Result	Pass/ Fail	Comment
1	Enter the email and password	The system checks the database for the account and let the user in to the home page.	As expected	Pass	
2	Navigate to projects.	Display a list of pre-created projects, if any, with the option to create a new project.	As expected	Pass	
3	Clicks on creating a new project.	Displays a form to fill in with the info about the project.	As expected	Pass	
4	Fill in the necessary info and click on Save.	Display a pop-up window for confirming the successful process	As expected	Pass	

Post-conditions:
A project is created successfully.

Test case #	4.2	Associated user case ID	UC09
Test designed by	Mohammed Al-mohannadi	Test design date	18/05/2023
Executed by	Mohammed Al-mohannadi	Execution date	18/05/2023
Test case name	Assign trainee to projects		
Short description	Specify a project for a trainee.		

Pre-conditions:
- There are trainees available to sign up.
- The user is logged into the system.
- There are available projects to sign with.

Step	Test Step	Expected System Response	Actual Result	Pass/ Fail	Comment

1	Enter the email and password	The system checks the database for the account and let the user in to the home page.	As expected	Pass	
2	Navigate to projects.	Display a list of pre-created projects.	As expected	Pass	
3	Clicks on one of the projects.	Displays information about the project with the option to add and remove trainees.	As expected	Pass	
4	Click add trainee.	Display a list of qualified trainees	As expected	Pass	
5	Click on the wanted to assign trainee and click Add button	Send a notification to the assigned trainees.	As expected	Pass	
Post-conditions: **The trainee is signed into a project successfully.**					

Other Appendices

Appendix C – Flutter Resources

Practice: -

- Flutter Tutorials (flutter.io)
- Flutter & Dart - The Complete Guide (www.udemy.com)
- Flutter in Action (www.manning.com)

Videos: -

- Iowwhu#wxwruldo#ru#ehjlqqhuv#lq#5#krxuv=#kwwsv=22z z z 1|rxwxeh1frp 2z dwfkBy@F 0 iND}qWuOX#
- P lqlp dd#Orjlq#XI#,#Ioxwhu#Wxwruldo# kwwsv=22z z z 1|rxwxeh1frp 2z dwfkBy@dMqImUlsjVn#
- Iowwhu#wxwruldo#ru#ehjlqqhuv#lq#1#krxu=#kwwsv=22z z z 1|rxwxeh1frp 2z dwfkBy@F 0 iND}qWuOX#

Books: -

- Gjlrssrsl#Fuu#Ij{jqtur jsy#| nym#Kqzyyjw
- Gjlrssrsl#Kqzyyjw#F#Mfsix#ts#Lzrij#t#Fuu#Ij{jqtur jsy
- Kqzyyjw#Fuuwjsyhj

Appendix D – Conducted Survey(s)

What is your gender? *

- ◯ Male
- ◯ Female

What is your current major? *

Choose

Were you ever a part of an internship program? *

- ◯ Yes
- ◯ No, not planning to do
- ◯ Would like to be a part of in the future

If you were a part of an internship program, was the experience worth it?

- ◯ Yes
- ◯ No
- ◯ Not that much

If you were a part of an internship program, Which of these problems did you face while doing an internship?

- [ ] Unfair amount of tasks
- [ ] Lack of communication between your employer and coordinator
- [ ] You were confused at times because of your employer
- [ ] Assigned to Tasks that were not related to your major
- [ ] Other: _____

If you were a part of an internship program, how easy was it to apply for an internship?

	1	2	3	4	5	
Very hard	○	○	○	○	○	Very easy

If you were a part of an internship program, How would you evaluate the benefits you've gotten from the internship?

	1	2	3	4	5	
No benefits ..	○	○	○	○	○	Extremely beneficial!

If you were a part of an internship program, Which of these problems did you face while doing an internship?

- [ ] Unfair amount of tasks
- [ ] Lack of communication between your employer and coordinator
- [ ] You were confused at times because of your employer
- [ ] Assigned to Tasks that were not related to your major
- [ ] Other: _____

If you were a part of an internship program, how easy was it to apply for an internship?

	1	2	3	4	5	
Very hard	○	○	○	○	○	Very easy

If you were a part of an internship program, How would you evaluate the benefits you've gotten from the internship?

	1	2	3	4	5	
No benefits ..	○	○	○	○	○	Extremely beneficial!

If you were a part of an internship program, Which of these problems did you face while doing an internship?

- [ ] Unfair amount of tasks
- [ ] Lack of communication between your employer and coordinator
- [ ] You were confused at times because of your employer
- [ ] Assigned to Tasks that were not related to your major
- [ ] Other: _____

If you were a part of an internship program, how easy was it to apply for an internship?

	1	2	3	4	5	
Very hard	○	○	○	○	○	Very easy

If you were a part of an internship program, How would you evaluate the benefits you've gotten from the internship?

	1	2	3	4	5	
No benefits ..	○	○	○	○	○	Extremely beneficial!

How easy was it to creat a new account or log in in to your existing account *

- ◯ Very easy and straightforward
- ◯ Somehow easy, but with minor difficulties
- ◯ Difficult and frustrating, encountered significant obstacles
- ◯ I did not attempt to create a new account or log in

Which of the following best describes the overall navigation experience in the application *

- ◯ Easy to navigate
- ◯ Average, could be improved in some areas
- ◯ Confusing and difficult to find what I need
- ◯ I haven't explored the navigation much

How would you rate the visual design and layout of the application *

- ◯ Visually appealing and well-organized
- ◯ Average, could be improved in some areas
- ◯ Unattractive and cluttered
- ◯ I haven't paid much attention to the design

How responsive was the application to your interactions (e.g., clicking buttons, loading times) *

- ◯ Very responsive, no noticeable delays
- ◯ Average, some minor delays but nothing significant
- ◯ Slow and laggy, affecting the user experience
- ◯ I haven't noticed any responsiveness issues

Which types of errors or unexpected behaviors did you encounter (Select all that apply) *

- ☐ Error messages without clear explanations
- ☐ Broken links or missing content
- ☐ Unresponsive buttons or actions
- ☐ Inconsistent behavior across different screens
- ☐ Slow loading times for specific features/pages
- ☐ Every thig is good

Please select the devices/platforms on which the application has been tested and found to be compatible: (Select all that apply) *

- [ ] Desktop/Laptop computer
- [ ] Mobile phones (iOS)
- [ ] Mobile phones (Android)
- [ ] Tablets (iOS)
- [ ] Tablets (Android)
- [ ] Web browsers (Chrome)
- [ ] Web browsers (Firefox)
- [ ] Web browsers (Safari)
- [ ] Web browsers (Edge)
- [ ] Other..

How easy was it to search for internship based on your interests, Skiles and location *

	1	2	3	4	5	
Very easy	○	○	○	○	○	Difficult

How would you rate the overall usability of the application *

- ○ Excellent, it exceeded my expectations
- ○ Good, but there is room for improvement
- ○ Poor, it fell significantly short of my expectations
- ○ I haven't formed an opinion on the overall usability

Internship Application Acceptance Test

Is the application's user interface intuitive and easy to use *

- ◯ Yes, it is intuitive and easy to use
- ◯ It is somewhat intuitive, but could be improved
- ◯ No, it is not intuitive and requires improvements

Does the application meet your initial requirements and expectations *

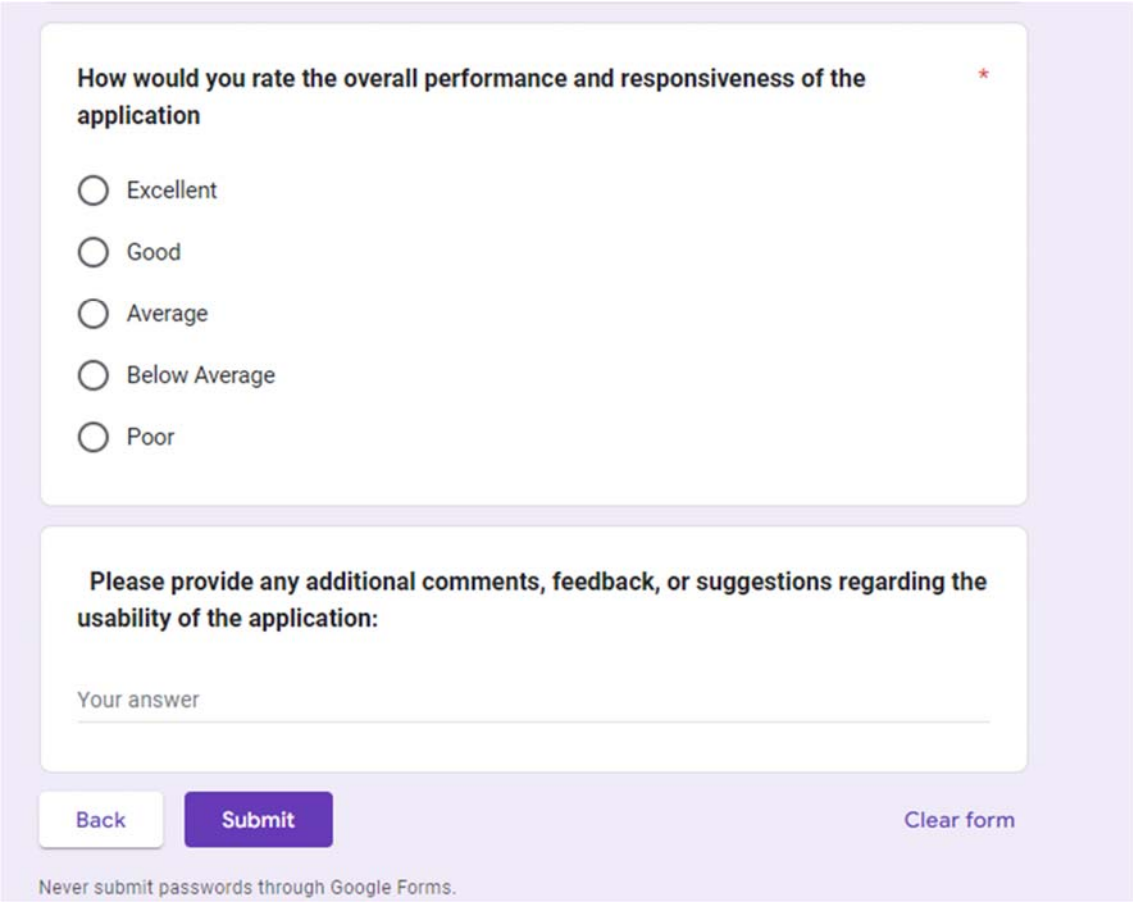
- ◯ Yes
- ◯ No
- ◯ Partially

If your answer is No, why

Your answer

How satisfied are you with the provided documentation and user guides *

- ◯ Very satisfied
- ◯ Somehow satisfied
- ◯ Not satisfied


How would you rate the overall performance and responsiveness of the application *
Excellent
Good
Average
Below Average
Poor
Please provide any additional comments, feedback, or suggestions regarding the usability of the application:
Your answer
Back
Submit
Clear form
Never submit passwords through Google Forms.

Appendix E – Meetings Conducted

Internship program

We had a private planned face to face interview meeting with Dr. Noora Fetais on 7/09/2022 from 8:00am to 9:00 am. She is the one responsible for providing students with the practical training programs in the computer science and engineering department. We discussed with her the important **system requirements** that they requested. She explained to us how slow and manual this process gets, and she wanted that process to be automated using a software developed by our team, the software should provide services for students to enable them to see the available training programs provided by companies. And what is the gained outcomes that students going to have after the completion of that program, the software also should assure that the training is provided for students who completed a certain number of credit hours or certain courses , she also mentioned some parts of the use case design where the system should be intermediate between the student , the university department and the company providing the training , where those are going to be the main actors of our system In conclusion after that meeting we adopted this project as our senior project and we are willing to design and develop a mobile application that is going to provide the mentioned services to facilitate the process of registering into internships.

Two groups meeting

We had a planned meeting with group number 11 on 25/09/2022 from 8:00am to 9:00 am in the BCR hallways in Qatar University. That meeting was extremely important because it was going to affect the future of this project and he we planned things out. The meeting started with us introducing ourselves and our ideas about the topic, so we can agree if a merger between our groups will happen. We talked about the framework we were going to use, which database will be applied and the whole structure of the project. After discussing for a while, we finally agreed to become a group of 6 members in total, and 2 supervisors. We both had the same goals we wanted to achieve, the same ambition and discipline. Everything went in the best direction and both this meeting and the merger of groups were very fruitful.