



Palestine
Polytechnic
University

Palestine Polytechnic University
College of Information Technology and Computer Engineering
Graduation Project

“Training Management System – Jawwal Case”

Project Team:

Raghad AbuSharar

Nagham Hejja

Ahmad Arafeh

Under Supervision Of

Dr. Radwan Tahboub

This research was presented to fulfill the requirements of the graduation project in the specialization of Information Technology and Computer Science at the College of Information Technology and Computer Engineering.

2021-2022

إهداء

لله الذي هدانا لهذا وما كنا لنهتدي لولا ان هدانا الله الحمد

إلى من لا توفيهـم الكلمات والحروف حقهم في البر والإحسان, إلى من رضا الله في رضاهم وما توفيقنا وسر نجاحنا إلا به م
إلى من كللهم الله بالهيبـة والوقار. إلى من علمونا العطاء بدون انتظار.. إلى من نحمل أسماءهم بكل افتخار. نرجو من الله ان يمد بأعماركم
لتروا ثماراً قد حان قطافها بعد طول انتظار.. وستبقى كلماتكم نجوما نهتدي بها اليوم وفي الغد والى الابد

والدينا العزيزين

إلى من رافقونا طوال السنين وشاركونا الأفراح والألام.. إلى معنى الحب و الحنان والتفاني .. إلى بسمة الحياة وسر الوجود.. إلى من كان
دعائهم سر نجاحنا وحنانهم سر تقدمنا إلى أمهاتنا إلى من هم أقرب إلينا من روحنا إلى من شاركونا حضن الأم وبهم نستمد عزتنا و

إصرارنا

إخوتنا الأعزاء

إلى من أنسنا في دراستنا وشاركنا همومنا تذكراً و تقديرأ

أصدقائنا

وإلى من ساندنا وشجعنا طوال مسيرتنا الدراسية

جميعا نهدي ثمرة هذا الجهد المتواضع وفاءأ إلى حاملي لواء النور والسائرين في دربهم بإخلاص كل أساتذتي وجميع طلاب العلم إليهم

وتقديرأ و عرفانأ له م

شكر وتقدير

بسم الله الرحمن الرحيم

نتوجه بالشكر والتقدير إلى جامعة بوليتكنيك فلسطين التي احتضنتنا لإكمال دراستنا وحصولنا على درجة البكالوريوس. وإلى الهيئة التدريسية في كلية تكنولوجيا المعلومات وهندسة الحاسوب. وقبل أن نمضي نقدم أسمى آيات الشكر والامتنان والتقدير والمحبة إلى مشرفنا الدكتور رضوان طهبوب الذي تفضل بالإشراف على هذا البحث فجزاه الله كل خير وله منا كل التقدير والاحترام على جهوده الطيبة ومتابعته التي بذلها في سبيل تعليمنا ونجاحنا, وأيضا نتوجه بالشكر إلى شركة جوال ممثلة بالأستاذ فيصل شحادة والأستاذ محمد نصار والأستاذ أيمن غانم والأستاذة رنا النصر على ما بذلوه معنا من جهود قيمة, وإلى كل من ساهم في انجاز هذا المشروع

ملخص المشروع

المشروع عبارة عن تطبيق هاتف ذكي لتنظيم العملية التدريبية بين الأطراف الأربعة وهم: المتدربين ، المدربين ، الشركة ومشرف التدريب الميداني الجامعي. ويهدف التطبيق إلى مساعدتهم على الحصول على تجربة تدريبية سهلة ومنظمة تنظيماً جيداً. يمكن للمتدربين تتبع مهامهم و وقت القدوم والمغادرة لحساب مجموع الساعات فور الانتهاء من الساعات المطلوبة من الطالب المتدرب وكتابة التقارير اليومية بالمهام التي قاموا بتطبيقها أو تعلمها. كما يمكن للمدربين إضافة المهام وتكليفها للمتدربين وتقييم أداء كل متدرب لإعداد التقارير. تعتبر الشركة كمسؤول عن النظام ، ويمكنها إدارة المستخدمين من خلال صفحة الويب عن طريق قبول طلبات التدريب أو رفضها وإضافة المستخدمين وتعيين مدرب لكل متدرب. بالنسبة لمشرف التدريب الجامعي يمكنه متابعة أداء كل متدرب وعرض تقييمه و وقت قدومه و مغادرته من الشركة بالإضافة الى عرض الأشياء التي قام بتعلمها وتطبيقها وعرض قائمة الطلاب الذين انهوا ساعاتهم التدريبية

Abstract

The project is a smart mobile application for organizing the training process between the 4 parties, which are: Trainees, Trainers, Company, and University Training Supervisor, it aims to help them having an easy and well-organized training experience. University Students (Applicants) will apply for the required form, after they are accepted, they will be a part of the system as a Trainees so they can track their tasks, and add enter their coming and leaving time. they also can write daily reports with things they have learned or applied. Trainers can add tasks and assign them to trainees. They can also evaluate each trainee's performance to generate reports for them. The company is considered as an admin for the system, it can manage users through the admin panel web page by accept or reject applicant forms and assign each trainee to its trainer. For the University Training Supervisor, he can track the performance of each trainee and view their evaluation, coming and leaving times and the thing they gained from the training they also can view a list of the Trainees who finished the required training hours.

Table of Contents

Introduction.....	9
1.1 Overview:	10
1.2 Motivation:	10
1.2.1 For trainees:	10
1.2.2 For trainers:.....	10
1.2.3 For the university training supervisors:	10
1.2.4 For the Company:	11
1.3 Scope of the system:	11
1.4 Objectives:	11
1.5 Importance of the system:.....	12
1.5.1 For trainees:	12
1.5.2 For trainers:.....	12
1.5.3 For university training supervisor:.....	12
1.5.4 For company:	13
1.6 Procedure to achieve:.....	13
1.7 Short description of the system:.....	13
1.7.1 For the Applicant student:.....	13
1.7.2 For trainee:.....	14
1.7.3 For the trainer:	14
1.7.4 For the university training supervisor:	14
1.7.5 For the company:	15
1.8 Context diagram:	15
1.9 Conclusion:	16
Requirements specification.....	17
2.1 Introduction:	18
2.2 Functional requirements:	18
2.2.1 Company functional requirements:.....	18
2.2.2 Trainers functional requirements:	18
2.2.3 Trainees functional requirements:	18
2.2.4 University training supervisor functional requirements:	19
2.3 Non-functional requirements:	19
2.3.1 Usability:.....	19
2.3.2 Maintainability:.....	19
2.3.3 Scalability:	19
2.3.4 Portability:	19
2.3.5 Security:.....	19
2.4 Domain requirements:	20

2.4.1 Company domain requirements:	20
Software design	30
3.1 Introduction:	31
3.2 Use case diagram:	31
3.3 Sequence diagrams:	32
3.6 Block diagram:.....	36
3.4 Data flow diagram:	37
3.5 Class diagram:	38
3.6 Database mapping:.....	39
3.8 User interfaces:	47
3.8.1 Admin panel:	47
3.8.2 Mobile Training Application	49
Software demonstration	52
4.1 Introduction:	53
4.2 Software needed for the development stage:	53
4.3 System programming:.....	54
4.3.1 The mobile application:	54
4.4 The web application:.....	55
4.4.1 Backend Node.js:.....	55
4.4.2 PhpMyAdmin:	55
Testing	56
5.1 Introduction:	57
5.2 Validation:	57
5.3 Screen testing:.....	57
5.4 API request testing:.....	57
5.4.1 Log-in validations:.....	59
Conclusion	61
6.1 Recommendations:	62
6.2 Future work:.....	62
6.3 References:	62
6.3.1 Sites used in developing the system:	62

List of figures

Figure 1.8.1 Context diagram	16
Figure 3.2.2 Use case diagram	32
Figure 3.3.3 Accept/reject applicants sequence diagram	32
Figure 3.3.4 Sign in sequence diagram	33
Figure 3.3.5 Add user sequence diagram	33
Figure 3.5.7 Delete user sequence diagram	34
Figure 3.3.6 Generate reports sequence diagram	34
Figure 3.3.7 Download reports/certificates sequence diagram	35
Figure 3.3.8 Recording coming and leaving time sequence diagram	35
Figure 3.6.9 Block diagram	36
Figure 3.4.10 Data flow diagram	37
Figure 3.5.11 Class diagram	38
Figure 3.6.12 Database mapping	39
Figure 3.8.1.13 Admin panel: applied forms page	48
Figure 3.8.1.14 Admin panel: trainees list page	49
Figure 4.2.25 Software used in development	55
Figure 4.3.1.3.26 MVVM model	56
Figure 5.4.1.27 Login Validations(required fields)	59
Figure 5.4.1.28 Login validations(invalid email)	59
Figure 5.4.1.29 Login validations(invalid password)	60

List of Tables

Table 2.4.1.1 Create a list of trainers.....	20
Table 2.4.1.2 Accept or reject trainees	21
Table 2.4.1.3 Assign each trainee to its trainer	21
Table 2.4.1.4 Add or delete users.....	22
Table 2.4.1.5 Generate a certificate for each trainer	23
Table 2.4.2.6 Trainer sign in	23
Table 2.4.2.7 Create tasks list	24
Table 2.4.2.8 Confirm trainees training hours	24
Table 2.4.2.9 Generate reports and evaluations for trainees	25
Table 2.4.3.10 Trainee sign in	25
Table 2.4.3.11 Recording coming and leaving hours	26
Table 2.4.3.12 View tasks list.....	26
Table 2.4.3.13 Filling a report with what trainee has learned	27
Table 2.4.4.14 University training supervisor sign in	27
Table 2.4.4.15 Track trainees' performance	28
Table 2.4.4.16 Downloading reports and certificates	28
Table 2.4.4.17 View trainees who finished training.....	29
Table 3.7.18 Database tables	40
Table 3.7.19 Application form table	41
Table 3.7.20 Trainer table	42
Table 3.7.21 Company table.....	43
Table 3.7.22 University training supervisor table	44
Table 3.7.23 Trainee table	45
Table 3.7.24 Tasks list table	46
Table 3.7.25 Employee table	46
Table 3.7.26 Available Hours table	47
Table 5.4.26 User sign-up	59
Table 5.4.27 User log-in	59

Chapter 1

Introduction

- **Overview**
- **Motivation**
- **Scope of the system**
- **Objectives**
- **Importance of the system**
- **Procedure to achieve**
- **Short description of the system**
- **Context diagram**
- **Conclusion**

1.1 Overview:

At some point, university students have to try a real work experience in their field of study, so they have to look for a company that suits their major and start their training, otherwise, the university itself may send students to some specific companies to train. There's a difference from one university to another according to their policy. This project aims to develop a mobile application for trainers, trainees, the training company, and the university training supervisor to reflect the college approach in using technologies to support the training process. The application will facilitate the training process that may be somehow complicated at some point for all parties, it will include many features that will help in the training process.

1.2 Motivation:

Through the experience and viewing the research team on how the training process occurs between universities and companies, we found that there are many difficulties faced by trainees, trainers, the company, and the university training supervisor at the same time.

1.2.1 For trainees:

- They have a problem with keeping up to date with the tasks that have to be done.
- They may have some difficulties with calculating their training hours.
- They spend much effort when sending the university request training for the company as well as sending the company evaluation to the university training supervisor.

1.2.2 For trainers:

- They have difficulties with tracking a big number of trainees.
- They don't make sure that all trainees are up to date with their tasks.
- They have difficulties with tracking trainees' attendance.

1.2.3 For the university training supervisors:

- They have difficulties with keeping up to date with their student's attendance and performance in training.
- They don't make sure if their students have benefited from their training.

- They have a problem with making sure that all students have finished their required training hours or not.
- They have pressure with meeting every single student at the end of the semester to figure out what they have gained from their training.
- the piling of students' training papers at the end of the semester.

1.2.4 For the Company:

- It has a problem of managing a big number of trainees and accepting them.
- It may have difficulties with assigning trainees to trainers and making sure that every trainee has a trainer.
- It sometimes has problems with managing the training process in general from trainers to trainees when adding or deleting them.

Accordingly, it was suggested to develop a mobile application that will make a difference in the training field and help the four parties in managing the training process by eliminating all the difficulties they face as much as possible. We chose Jawwal Company as a case study for our project to make the ideas clearer and more specific.

1.3 Scope of the system:

The system will target university trainees, the company itself, company trainers, and the university training supervisor.

1.4 Objectives:

1. To make trainees have a better and easy training procedure by following up their training time and the tasks performed or have to be performed.
3. Enable the trainee to calculate his total hours easily.
4. Give trainers an easy way to track their trainee's attendance and training hours.
5. Enable the trainer to give the required trainees' tasks easily.
6. Enable trainers to evaluate trainees easily in the system by generating reports.

7. Enable the university training supervisor to track their student's attendance and performance in an easy way instead of making a phone call or sending an email to the company by viewing each trainee student profile anytime they want.
8. Make the university training supervisor sure that all their students finished training hours with a good performance.
9. Give the company an easy way to manage trainers and trainees by adding or deleting them and assigning trainees to trainers.

1.5 Importance of the system:

The importance of the system focuses on developing a tangible solution for the difficulties of the training process by making it easier for the four parties to deal with this process. The importance of the system can be summarized as the follows:

1.5.1 For trainees:

1. Keeping following up with their tasks.
2. they will be able to make daily reports about gained skills and tasks they have learned.
3. It will be easier for them to calculate their total training hours.

1.5.2 For trainers:

1. The application will facilitate performing the trainees with the required tasks.
2. They will be able to track Trainee's attendance easily.
3. Making reports and evaluations for trainees in an easy way.

1.5.3 For university training supervisor:

1. Tracking the trainee performance and schedules will be easier.
2. They can view all of this information on the application instead of making a phone call to the company.
3. It will give an easy way to have the reports and evaluation for each trainee student.

1.5.4 For company:

1. It will be easier for them to manage the big number of trainees and accept them.
2. Facilitate assigning trainees to their trainers.
3. Giving an easy way to add new trainees or trainers or delete them.

1.6 Procedure to achieve:

The research team relied on the software engineering process which begins with the planning of the system, then analysis of requirements followed by the design of the system, then development operation, and finally testing. We will also rely on agile methodology.

1.7 Short description of the system:

The system contains 4 parties:

1. The trainee
2. The trainer
3. The university training supervisor
4. The company

1.7.1 For the Applicant student:

The first page is shown when the applicant opens the application. It contains the following:

1. A form to fill with the following:

- Full name
- Phone number
- ID Number
- Address
- Email
- University

- Field
- Expected day of graduation
- Total average
- The training is mandatory or optional
- Days and hours available for each day
- Required training hours
- Training supervisor phone number
- Attachment for the university request to train with the training requirements
- Corona vaccination certificate

After submitting the form, the company will contact the user for an interview, after that he will get an email if he is accepted or not. When the trainee's application form is accepted the system will generate an account for this trainee so he can log in to the system with his credentials.

1.7.2 For trainee:

1. He will sign-up then sign-in to the application.
2. There will be a report page to record the topics and skills he will gain during his training period.
3. There will be a form page with coming and leaving time hours, then after trainer approval to these times, the total hours will be calculated.

1.7.3 For the trainer:

1. He will sign-up then log-in to the application.
2. Create a to-do list for trainees and he can add or delete the tasks.
3. At the end of the training period, the trainer will evaluate each trainee with a report and give a certificate from the company

1.7.4 For the university training supervisor:

1. He will sign-up then log-in to the application.
2. He will have access to the trainee's profiles to track their attendance and performance anytime they want.

3. He also can view the evaluation reports that the trainer will generate and the company certificate and download them.
4. He will view a list of students who finished their training hours.

1.7.5 For the company:

1. It will sign-up then log-in to the application
2. The company will accept or reject trainees who applied for the training.
3. It will create a list of trainers and assign trainees to them.
4. It also will be able to deactivate trainees accounts .
5. At the end of the training period, it will generate a certificate for each trainee.

1.8 Context diagram:

It shows the interactions between a system and the actors.

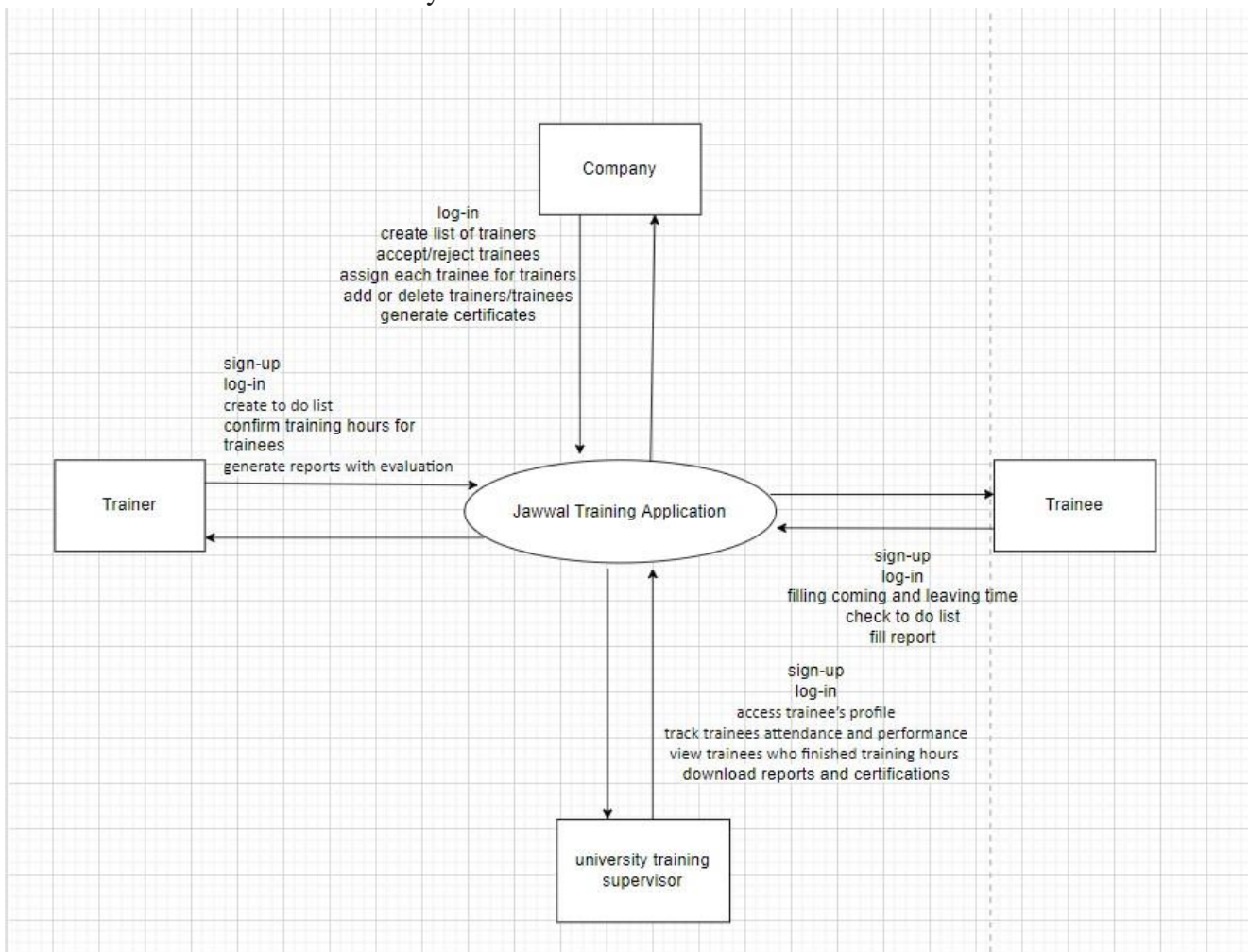


Figure 1.8.1 Context diagram

1.9 Conclusion:

In this chapter, we have talked about the research problem and solution, the objectives of the system, and how it is important. This introduction will be the basis of starting to talk about the second chapter which includes the requirements specifications.

Chapter 2

Requirements specification

- **Introduction**
- **Functional requirements**
- **Non-functional requirements**
- **Domain requirements**

2.1 Introduction:

This stage is one of the most important stages that the project is going through, at this chapter all Functional, non-functional, domain requirements will be clarified.

2.2 Functional requirements:

2.2.1 Company functional requirements:

- Create a list of trainers.
- Accept or reject trainees.
- Assign each trainee to its trainer.
- Add or delete users (trainees or trainers).
- Generate a certificate for each trainee.

2.2.2 Trainers functional requirements:

- Sign in to the personal account.
- Create tasks list.
- Confirm training hours that the trainee records.
- Generate reports with evaluation each day after approving training hours.

2.2.3 Trainees functional requirements:

- Sign in to the personal account.
- Record coming and leaving time.
- View and check the tasks list.
- Fill a report with what he has learnt and applied each day.

2.2.4 University training supervisor functional requirements:

- Sign in to the personal account.
- Access Trainee's profile and track their attendance and performance.
- Download the evaluation reports and certificates for trainees.
- View a list of trainees who finished their required training hours.

2.3 Non-functional requirements:

2.3.1 Usability:

- Removing ambiguity among users of the application, through an easy-to-use design interface, and access to all parts of the interface without obstacles, so that the information is presented in an easy and concise manner.
- Use eye-pleasing colors and avoid radioactive colors. We used green and white in our application.
- Clear language that serves the purpose of the application.

2.3.2 Maintainability:

- The design of the application will be able to modification and development based on Future changes and developments, by creating an easy and clear design linked to clear documentation.

2.3.3 Scalability:

- The application will be expandable to include new developments in the Training process, and adding new trainees to the application, it will absorb up to hundreds of users.

2.3.4 Portability:

- It's a responsive design for both web and mobile with different size screens

2.3.5 Security:

The Application provides protection from any external influences, as it is not possible to login unless the password and username are matched or when it authenticates the user by testing his role before log-in.

- Database security: Passwords were saved encrypted in the database so no one can access user accounts

- Login Authentication:
 - user can't login to their account unless they have their own email and password and the user status is activated.
 - users can only view pages unless they are authorized to access by the admin
- System security: the system will not check any password and email unless they have the correct data format in both signup and login pages.

2.4 Domain requirements:

After the project team collected information about the functional requirements of the application, it was divided into requirements related to the company, the trainee, the trainer and the university training supervisor.

2.4.1 Company domain requirements:

- Create a list of trainers

Table 2.4.1.1 Create a list of trainers

Job name	Create a list of trainers
The description	Enable the company to Create a list of trainers who will participate in the training process in the company
Input	Information of the trainer (name, id, field)
Output	A new trainer will be added to the list
Measures	<ol style="list-style-type: none"> 1. Open the trainer list 2. Click add new trainer 3. Fill trainer information 4. Click add

- Accept or reject trainees:

Table 2.4.1.2 Accept or reject trainees

Job name	Accept/reject trainees
The description	Enable the company to approve or reject the request based on the available information
Input	Approval based on the request
Output	Sending an email, making a phone call to give the final decision of approval or rejection and the interview details if the applicant was accepted
Measures	<ol style="list-style-type: none"> 1. Read the applicant's trainee information. 2. Approval or rejection of the application. 3. Conduct a personal interview with the trainee if he was initially approved.

- Assign each trainee to its trainer

Table 2.4.1.3 Assign each trainee to its trainer

Job name	Assigning trainees to trainers
The description	Enable the company to assign each accepted trainee to his trainer
Input	The name of the trainee
Output	A new trainee will be assigned to the trainer
Measures	<ol style="list-style-type: none"> 1. Open the trainer's list 2. Click assign trainee 3. Enter the trainee's name 4. Click assign

- Add or delete users (trainees, trainers, university training supervisor)

Table 2.4.1.4 Add or delete users

Job name	Add/delete trainees or trainers
The description	Enable the company to add trainers or trainees to the application when they're accepted or delete specific trainer or trainee when he stops or finishes training
Input	Information of the trainer/trainee that will be added or deleted(name, id)
Output	A new trainer/trainee will be added to the list or deleted from it
Measures	<p>Add:</p> <ol style="list-style-type: none"> 1. Open trainer/trainee lists 2. Click add new trainer/trainee 3. Fill trainer/trainee information 4. Click add <p>Delete:</p> <ol style="list-style-type: none"> 1. Open the trainer/trainee lists 2. click delete 3. Confirm deleting

- Generate a certificate for each trainee:

Table 2.4.1.5 Generate a certificate for each trainer

Job name	Accreditation certificate
The description	The company issues an accreditation certificate for the number of trainee training hours
Input	The name of the company, the university, the trainee, and the number of hours they have been trained
Output	A certificate approved by the company with the number of hours
Measures	<ol style="list-style-type: none"> 1. Enter the name of the trainee 2. Enter the name of the university 4. Record the number of training hours

2.4.2 Trainer domain requirements:

- Sign in to personal account:

Table 2.4.2.6 Trainer sign in

Job name	sign in
The description	Enable the trainer to access his account
Input	username and password
Output	open his personal account
Measures	<ol style="list-style-type: none"> 1. Open the sign-in page 2. Enter credentials 3. View personal account

- Create tasks list:

Table 2.4.2.7 Create tasks list

Job name	Add a new task
The description	Enable the trainer to add a new task for the trainee
Input	The task that the trainee must perform
Output	List of required tasks
Measures	<ol style="list-style-type: none"> 1. Add a new task 2. Modification of a previously added task 3. Delete or cancel the task

- Confirm trainee training hours:

Table 2.4.2.8 Confirm trainees training hours

Job name	Approval of training hours
The description	Enable the trainer to approve or reject training hours
Input	confirm or reject hours
Output	The hours of each day will be confirmed
Measures	<ol style="list-style-type: none"> 1. Ensure that the recorded training hours are consistent with the real hours 2. confirm training hours 3. rejection of the recorded hours if they do not agree with reality

- Generate reports and evaluations for the trainee:

Table 2.4.2.9 Generate reports and evaluations for trainees

Job name	Create a report to evaluate the trainee
The description	Enable the trainer to issue a report for each trainee
Input	Write a brief report for the trainee performance each day
Output	A report and evaluation by the trainer regarding the trainee performance
Measures	1. Fill out a brief and concise report for the work done by the trainee 2. Evaluate the trainee based on his work and achievement

2.4.3 Trainees domain requirements:

- Sign in to personal account:

Table 2.4.3.10 Trainee sign in

Job name	sign in
The description	Enable the trainee to access his account
Input	username and password
Output	open his personal account
Measures	1. Open the sign-in page 2. Enter credentials 3. View personal account

- Recording coming and leaving hours:

Table 2.4.3.11 Recording coming and leaving hours

Job name	Recording coming and leaving hours
The description	Recording working hours on a daily basis
Input	Day, date and Training start hour, and completion hour
Output	A form with training hours for each day
Measures	<ol style="list-style-type: none"> 1. Fill in the day, date, coming, and leaving time 2. submit

- View the tasks list

Table 2.4.3.12 View tasks list

Job name	Required tasks
The description	List of tasks to be performed by the trainee
Input	Open the required tasks page
Output	View Required tasks
Measures	<ol style="list-style-type: none"> 1. open the required tasks 2. Check the completed task

- Fill a report with what he has learned and applied each day

Table 2.4.3.13 Filling a report with what trainee has learned

Job name	Fill a report with achieved tasks for each day
The description	Enable the trainee to Create a daily report of what he has done and learned
Input	Writ the tasks he achieved and applied
Output	A report of what he has done every day
Measures	<ol style="list-style-type: none"> 1. Open the report page 2. Write the tasks he applied and the skills he acquired

2.4.4 University training supervisor domain requirements:

- Sign in to personal account:

Table 2.4.4.14 University training supervisor sign in

Job name	sign in
The description	Enable the University training supervisor to access his account
Input	username and password
Output	open his personal account
Measures	<ol style="list-style-type: none"> 1. Open the sign in page 2. Enter credentials 3. View personal account

- Access Trainee’s profile and track their attendance and performance:

Table 2.4.4.15 Track trainees’ performance

Job name	Track Trainees Performance
The description	Enable the supervisor to access the trainee’s profile
Input	Trainee’s Name
Output	View Trainee's Profile
Measures	<ol style="list-style-type: none"> 1. Enter the trainee’s name 2. View Trainee’s profile 3. View trainee’s attendance and performance

- Download the evaluation reports and certificates for trainees

Table 2.4.4.16 Downloading reports and certificates

Job name	Download the evaluation reports and certificates for trainees
The description	Enable the supervisor to download trainees’ certificates and reports
Input	Click download
Output	The report and certificate will be saved
Measures	<ol style="list-style-type: none"> 1. Open trainee profile 2. Open the report and certificate 3. Download

- View a list of trainees who finished their required training hours

Table 2.4.4.17 View trainees who finished training

Job name	View trainees who finished their required training hours
The description	Enable the supervisor to View the trainees who finished their required training hours
Input	Open the students who finished training hours page
Output	A list of trainees who finished training hours
Measures	<ol style="list-style-type: none"> 1. Open trainees who finished training hours 2. View the list

Chapter 3

Software design

- **Introduction**
- **Use-case diagram**
- **Sequence diagrams**
- **Block diagram**
- **Data flow diagram**
- **Class diagram**
- **Database mapping**
- **Database tables**
- **User interfaces**

3.1 Introduction:

This chapter includes an explanation of the design and structure of the project, where the components and the parts of the system will be detailed so that we give a Complete idea for all parts of the system. In terms of its design and internal components.

3.2 Use case diagram:

It explains the user’s interactions with the system, and what each user can do on the application like the following:

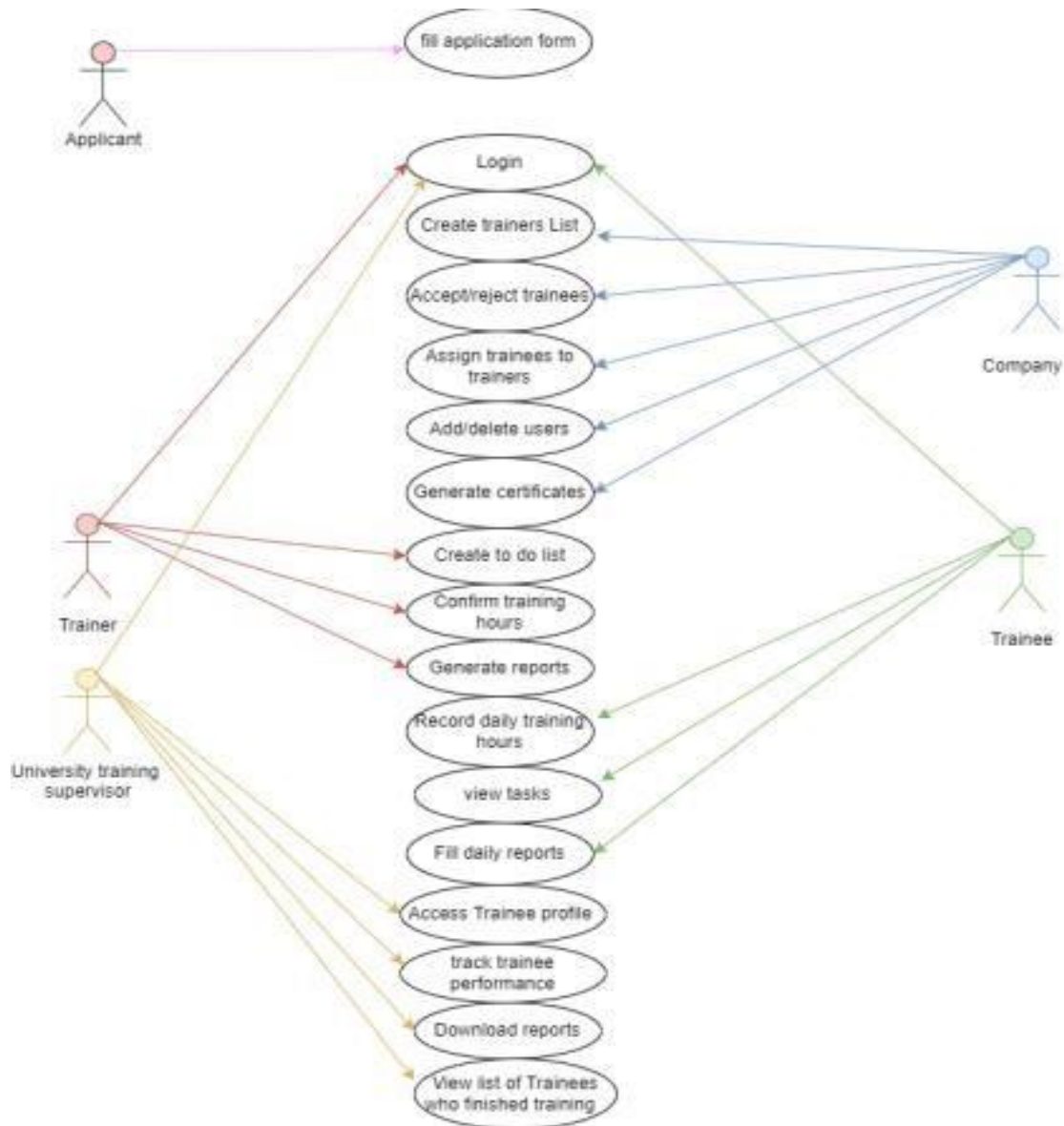


Figure 3.2.2 Use case diagram

3.3 Sequence diagrams:

It describes the main functions, how a specific operation is carried out in the application and how a group of the system objects works together:

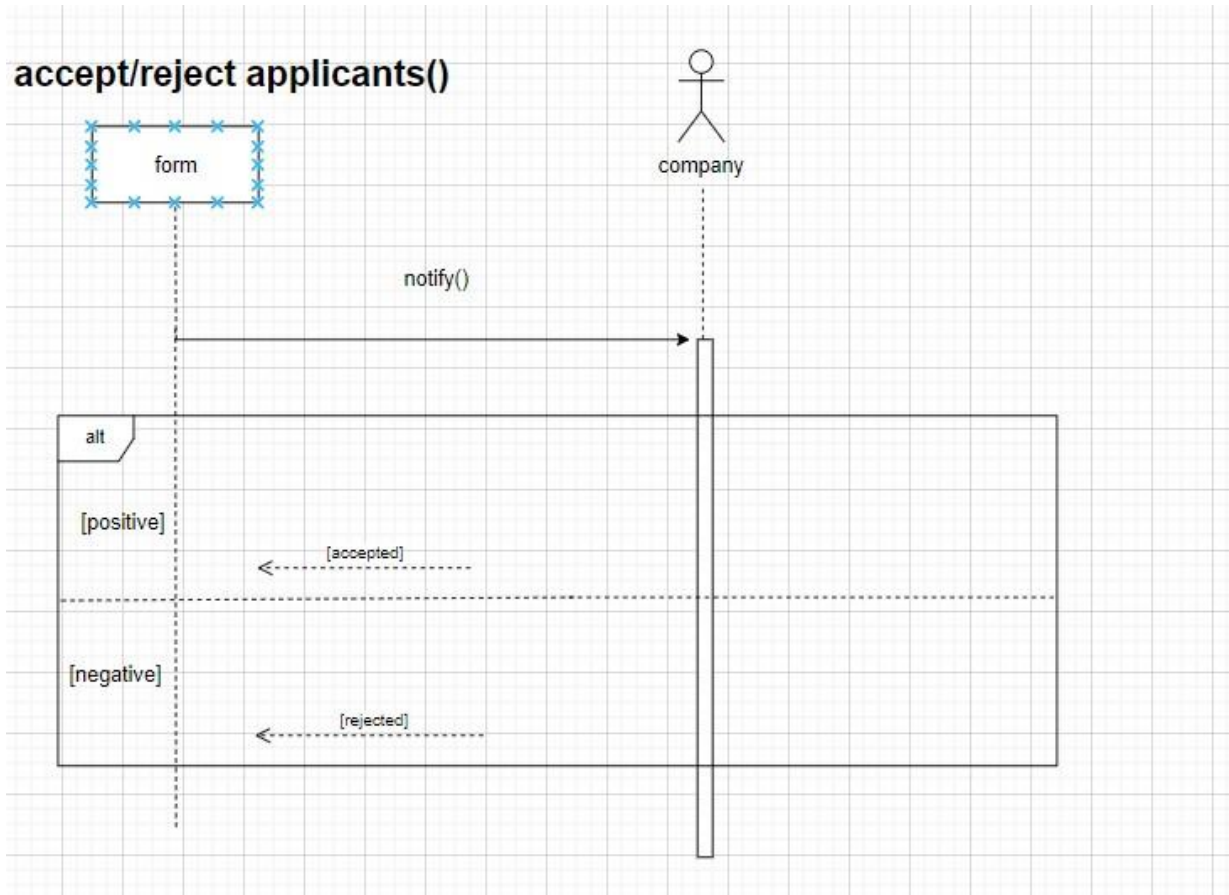


Figure 3.3.3 Accept/reject applicants sequence diagram

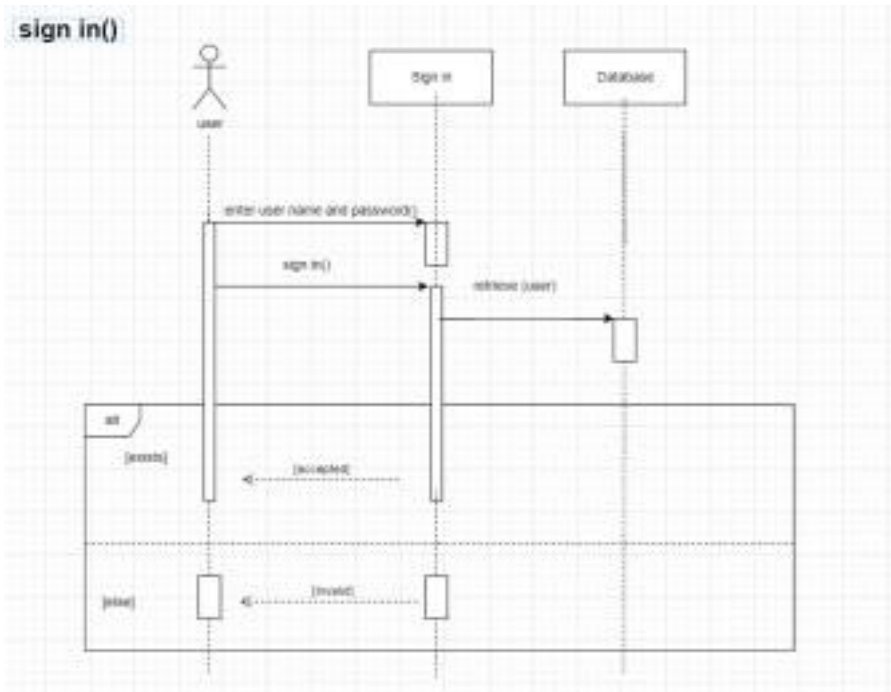


Figure 3.3.4 Sign in sequence diagram

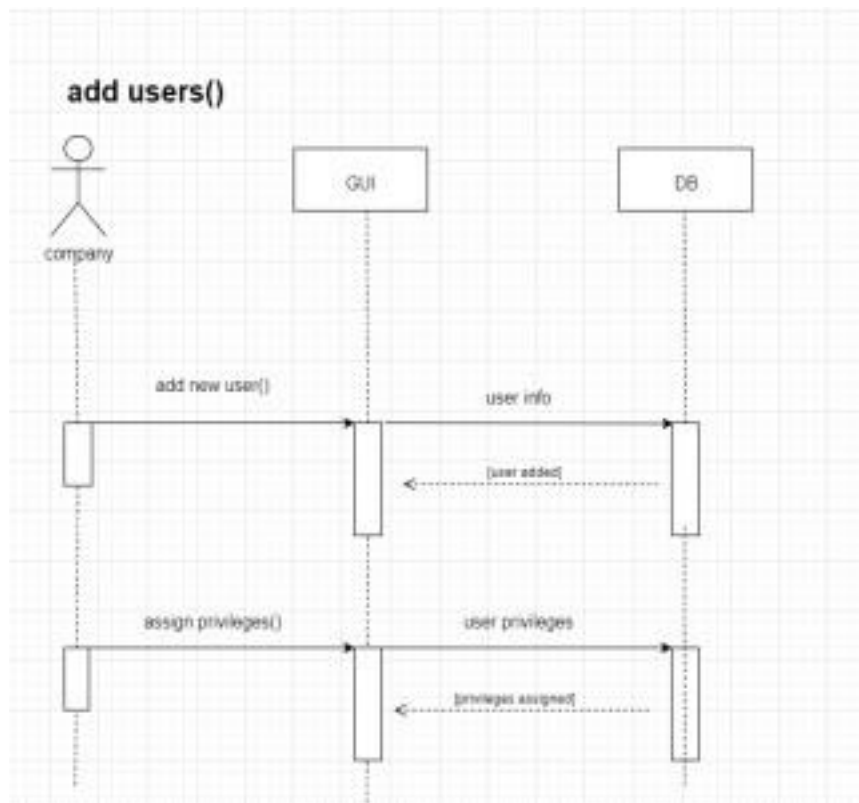


Figure 3.3.5 Add user sequence diagram

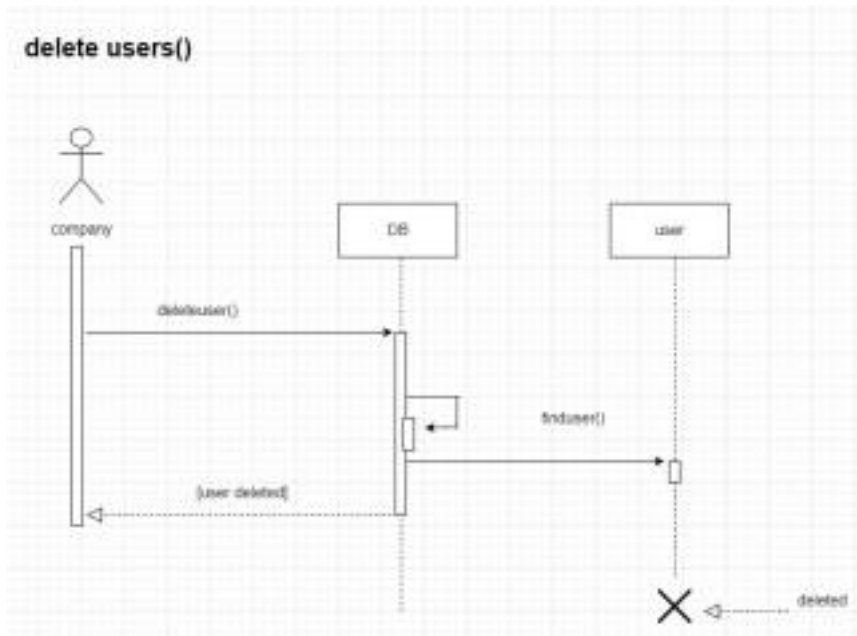


Figure 3.5.7 Delete user sequence diagram

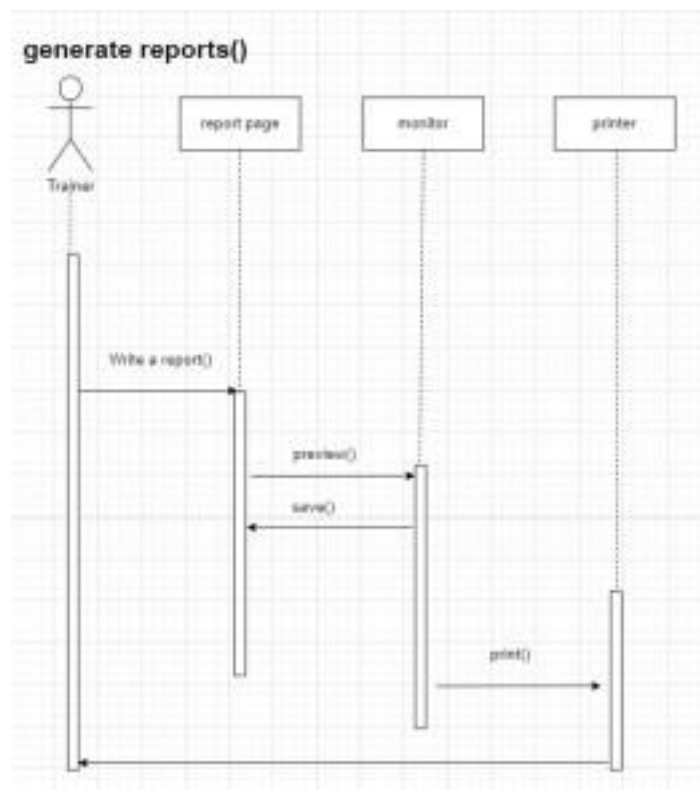


Figure 3.3.6 Generate reports sequence diagram

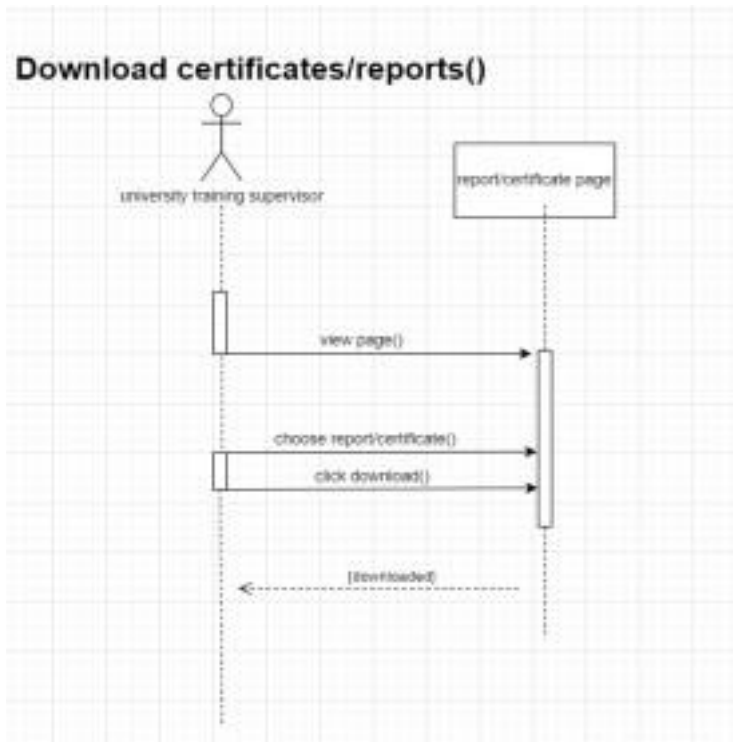


Figure 3.3.7 Download reports/certificates sequence diagram

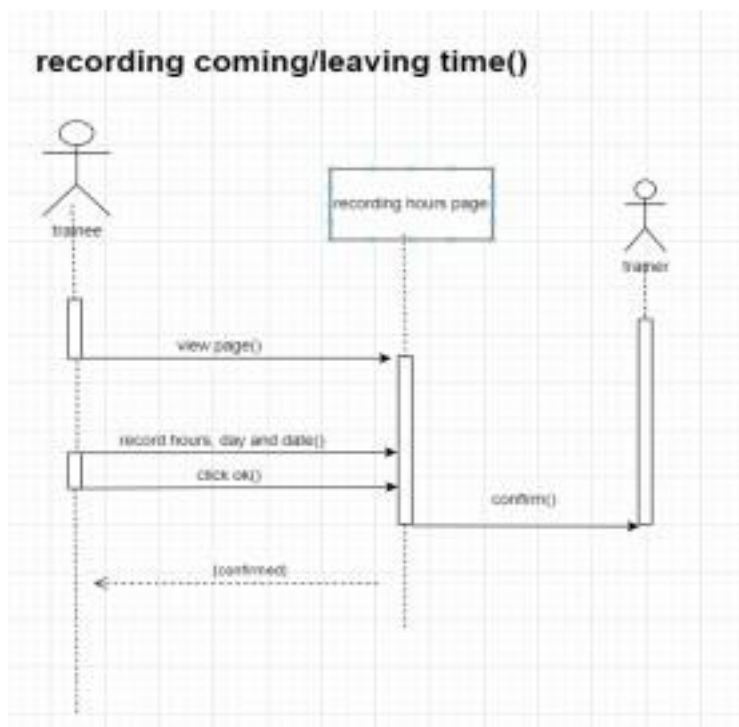


Figure 3.3.8 Recording coming and leaving time sequence diagram

3.6 Block diagram:

The following figure shows how the system works and deals with the database and users:

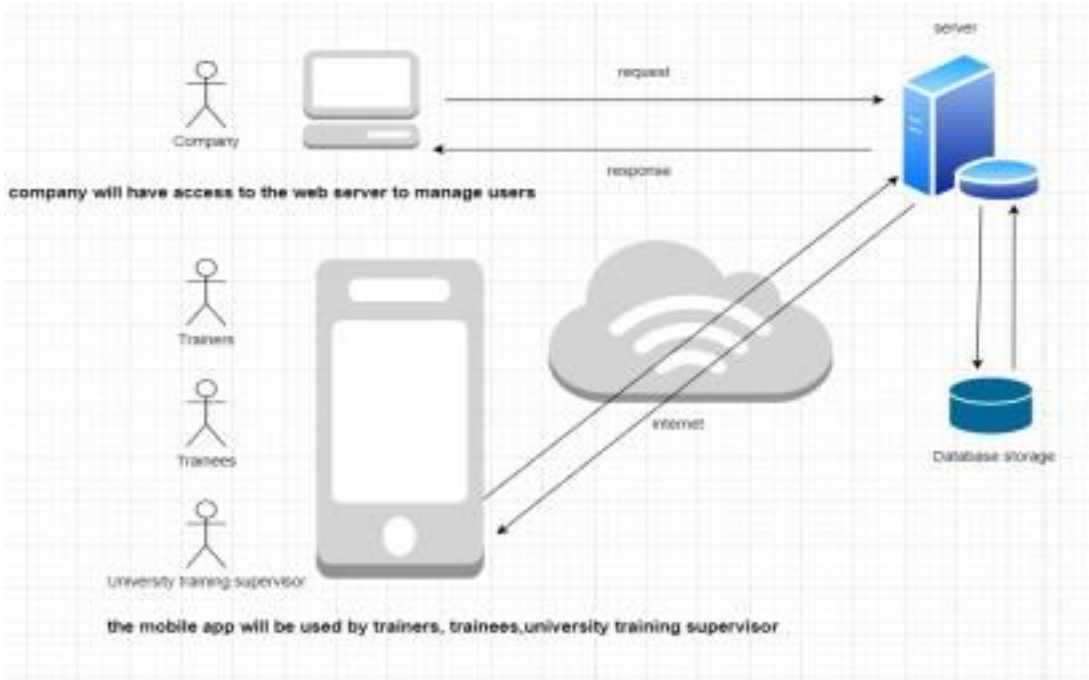


Figure 3.6.9 Block diagram

3.4 Data flow diagram:

The principal parts or functions are represented by blocks connected by lines that show the relationships of the blocks:

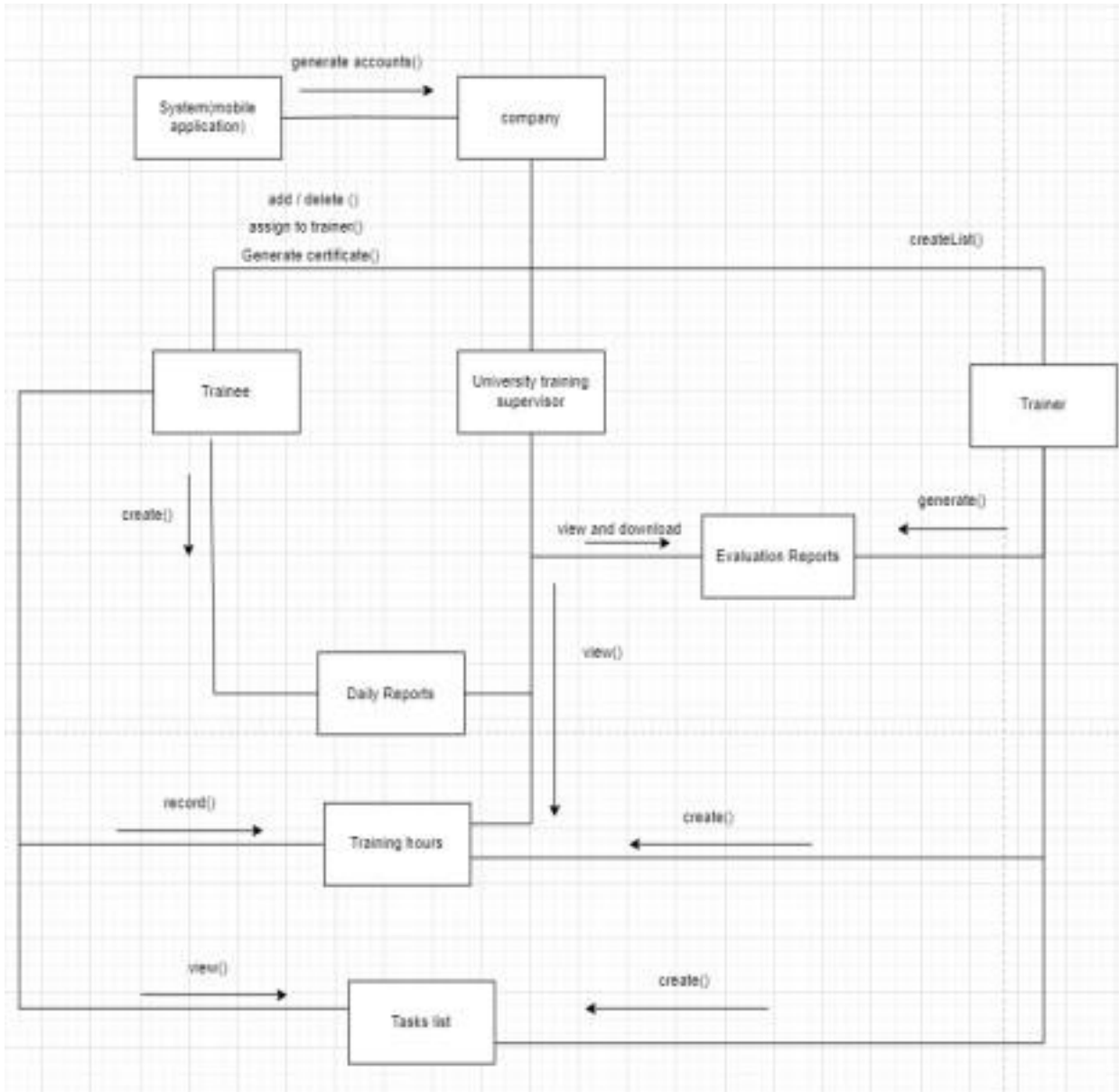
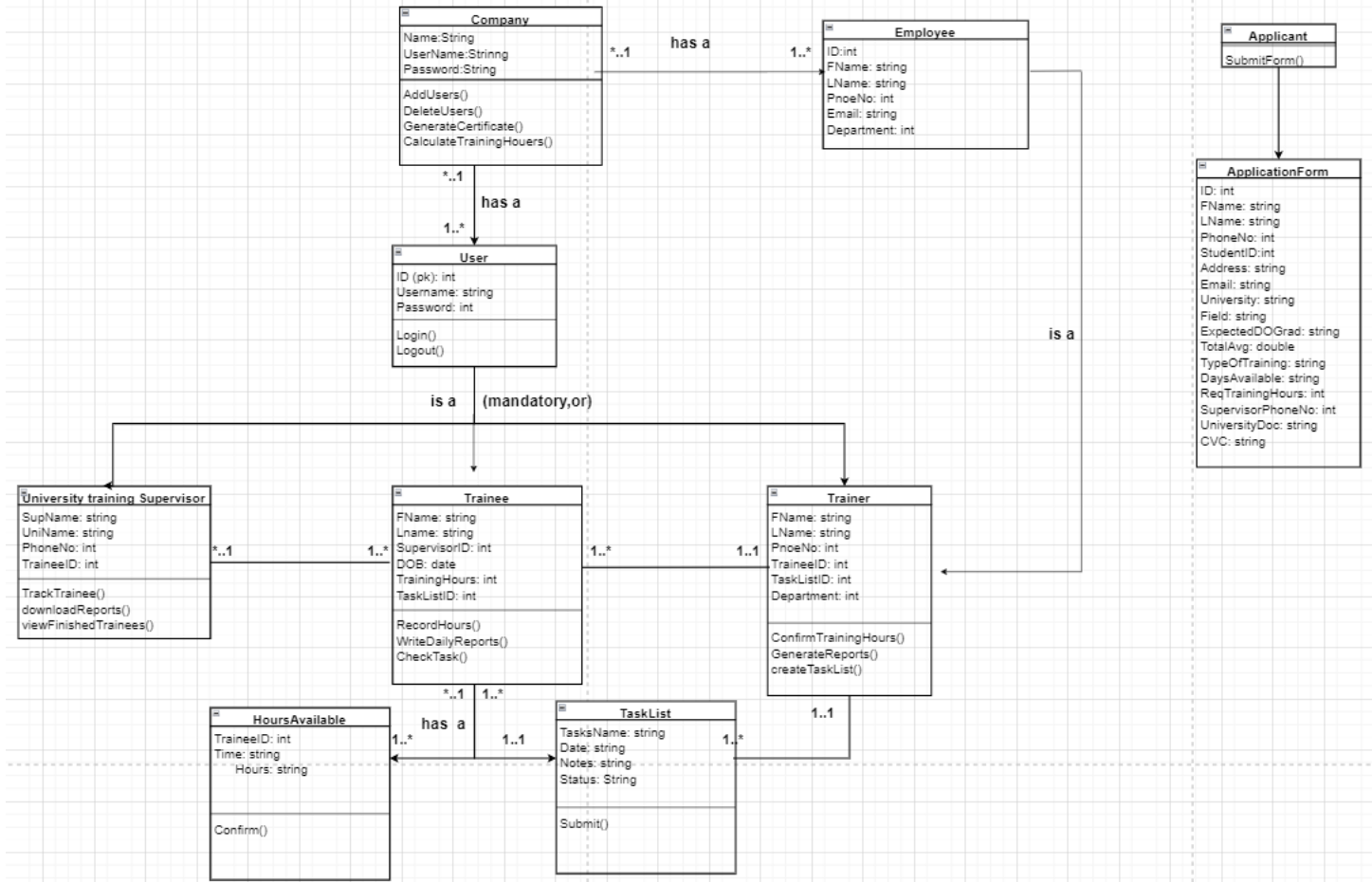


Figure 3.4.10 Data flow diagram

3.5 Class diagram:



It describes the structure of the system by showing the system's classes, attributes, operations and the relationships among objects:

Figure 3.5.11 Class diagram

3.6 Database mapping:

Determines database classes and relations between them:

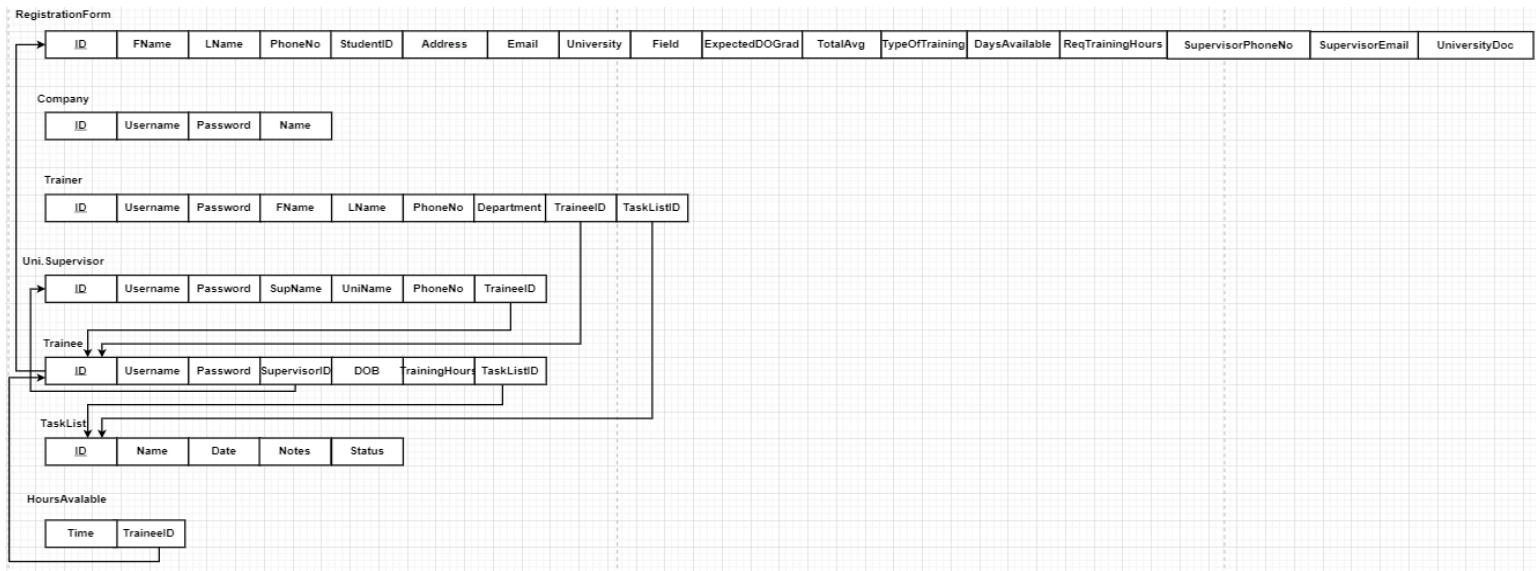


Figure 3.6.12 Database mapping

3.7 Database tables:

The system is associated with a database consisting of a number of tables interconnected with each other through common relationships between them as the following:

Table 3.7.18 Database tables

Table name	Description of tables
Application Form	Send the trainee's data.
Company	Full system data storage and control.
Trainer	Storing coach data, accessing the trainee, and following up on his performance and training hours.
University training supervisor	Store the university supervisor's data and access the trainees' performance and training hours
Trainee	Store trainee data and access training hours and task list.
Tasks List	Create a task list for trainees.
Hours Available	The hours which the trainee will be available to train

Application Form:

Table 3.7.19 Application form table

Field name	Field type	NULL	Field length	Field Description
ID	int	No	6	Primary key
FName	Varchar	No	10	Trainee's first name
LName	Varchar	No	10	Trainee's last name
PhoneNo	int	No	10	Trainee's phone number
Address	varchar	No	20	Trainee's address
Email	Varchar	No	30	Trainee's email
University	Varchar	No	30	University Name
Field	Varchar	No	10	Training field
Expected DOGrad	Date	Yes		Expected day of graduation
TotalAvg	int	No	10	Total average
TypeOfTraining	Varchar	No	10	Type of training (mandatory or optional)
DaysAvailable	Varchar	No	20	The days available for the trainee to practice
ReqTrainingHours	Time	No	10	Hours required of the trainee to complete
SupervisorPhoneNo	int	No	10	University supervisor phone number
Emailsup	Varchar	No	30	The training supervisor Email
UniversityDoc	BLOB	No		University request for training

Trainer:

Table 3.7.20 Trainer table

Field name	Field type	NULL	Field length	Field Description
ID	int	No	6	Primary key
Password	Varchar	No	10	Trainer profile password
FName	Varchar	No	10	Trainer's first name
LName	Varchar	No	10	Trainer's last name
PhoneNo	int	No	10	Trainer's phone number
TaskListID	int	No	6	Foreign key
TraineeID	int	No	6	Foreign key
Department	Varchar	No	15	The department in which the Trainer works

Company:

Table 3.7.21 Company table

Field name	Field type	NULL	Field length	Field Description
ID	int	No	6	Primary key
Username	Varchar	No	20	The username of the company to access the account
Password	Varchar	No	10	Company profile password
Description	Varchar	Yes	200	General description of the company
PhoneNo	int	No	10	Company contacts number
TrainerID	int	No	6	Foreign key
SupervisorID	int	No	6	Foreign key

University training supervisor:

Table 3.7.22 University training supervisor table

Field name	Field type	NULL	Field length	Field Description
ID	int	No	6	Primary key
Username	Varchar	No	20	The username of the supervisor to access the account
Password	Varchar	No	10	Supervisor profile password
SupName	Varchar	No	15	University supervisor name
UniName	Varchar	No	20	University Name
PhoneNo	int	No	10	Supervisor's phone number
TraineeID	int	No	6	Foreign key

Trainee:

Table 3.7.23 Trainee table

Field name	Field type	NULL	Field length	Field Description
ID	int	No	6	Primary key
Username	Varchar	No	15	The username of the trainee to access the account
FName	Varchar	No	10	Trainer's first name
LName	Varchar	No	10	Trainer's last name
Password	Varchar	No	10	Trainee profile password
SupervisorID	int	No	6	Foreign key
DOB	Date	No		Trainee's date of birth
TrainingHours	int	No	15	Training hours
TaskListID	int	No	6	Foreign key

TasksList:

Table 3.7.24 Tasks list table

Field name	Field type	NULL	Field length	Field Description
ID	int	No	6	Primary key
Name	Varchar	No	10	Task name
Date	Date	No		Task date
Notes	Varchar	No	200	Notes and details about the task
Status	Varchar	No	10	The percentage of finishing the task list

Employee

Table 3.7.25 Employee table

Field name	Field type	NULL	Field length	Field Description
ID	int	No	6	Primary key
FName	Varchar	No	10	Employee's first name
LName	Varchar	No	10	Employee's last name
PhoneNo	int	No	10	Employee's phone number
Email	Varchar	No	30	Employee's Email
Department	Varchar	No	15	The department in which the employee works

Table 3.7.26 Available Hours table

Field name	Field type	NULL	Field length	Field Description
TraineeID	int	No	6	Foreign key
Hours	Time	No	10	The hours available for the trainee to practice

3.8 User interfaces:

The screens that the users will interact with:

3.8.1 Admin panel:

- Applied forms page: this page will be controlled by the company; it shows the applicants forms and they can accept or reject it.

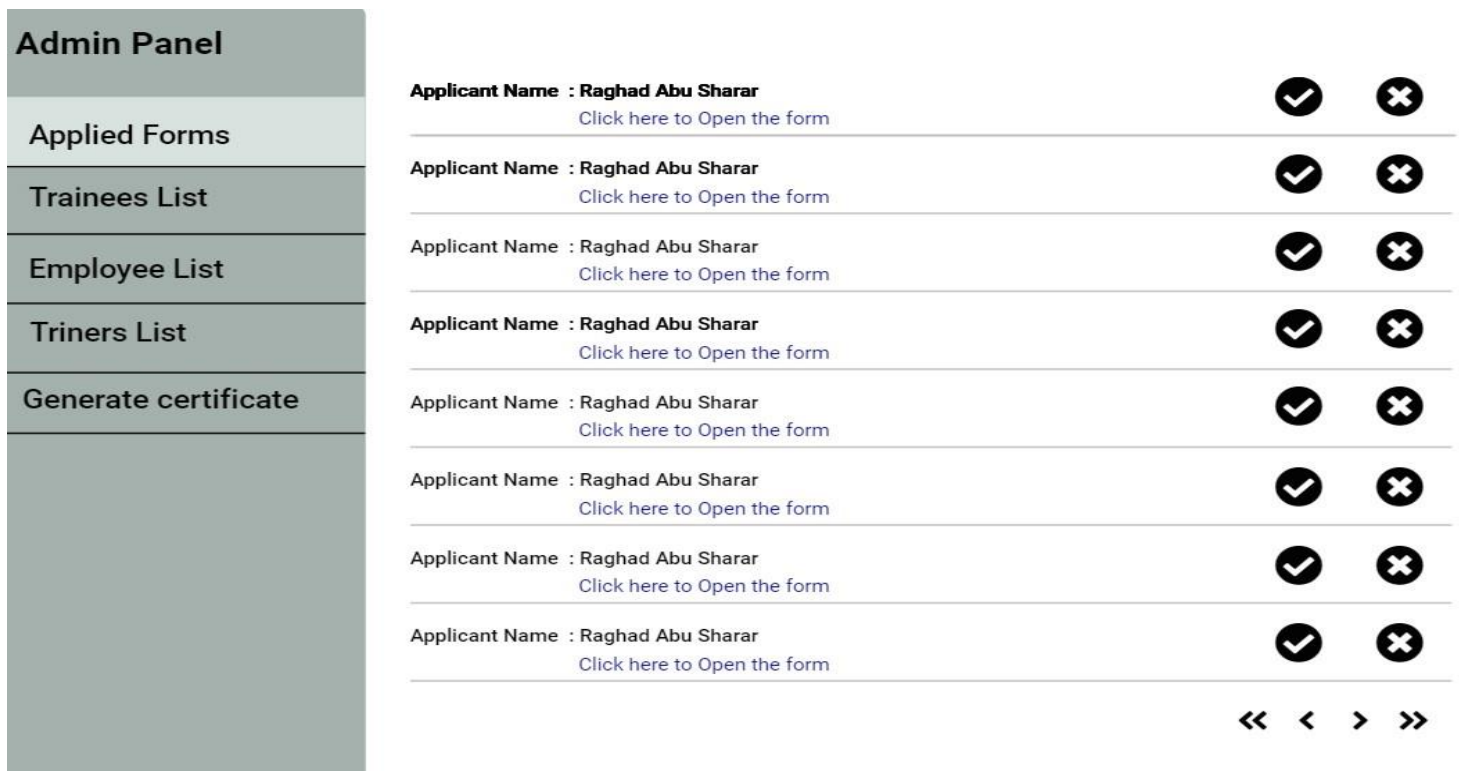


Figure 3.8.1.13 Admin panel: applied forms page

- **Trainees list page:** this page shows trainees list, with delete trainee and assign to trainer's options.

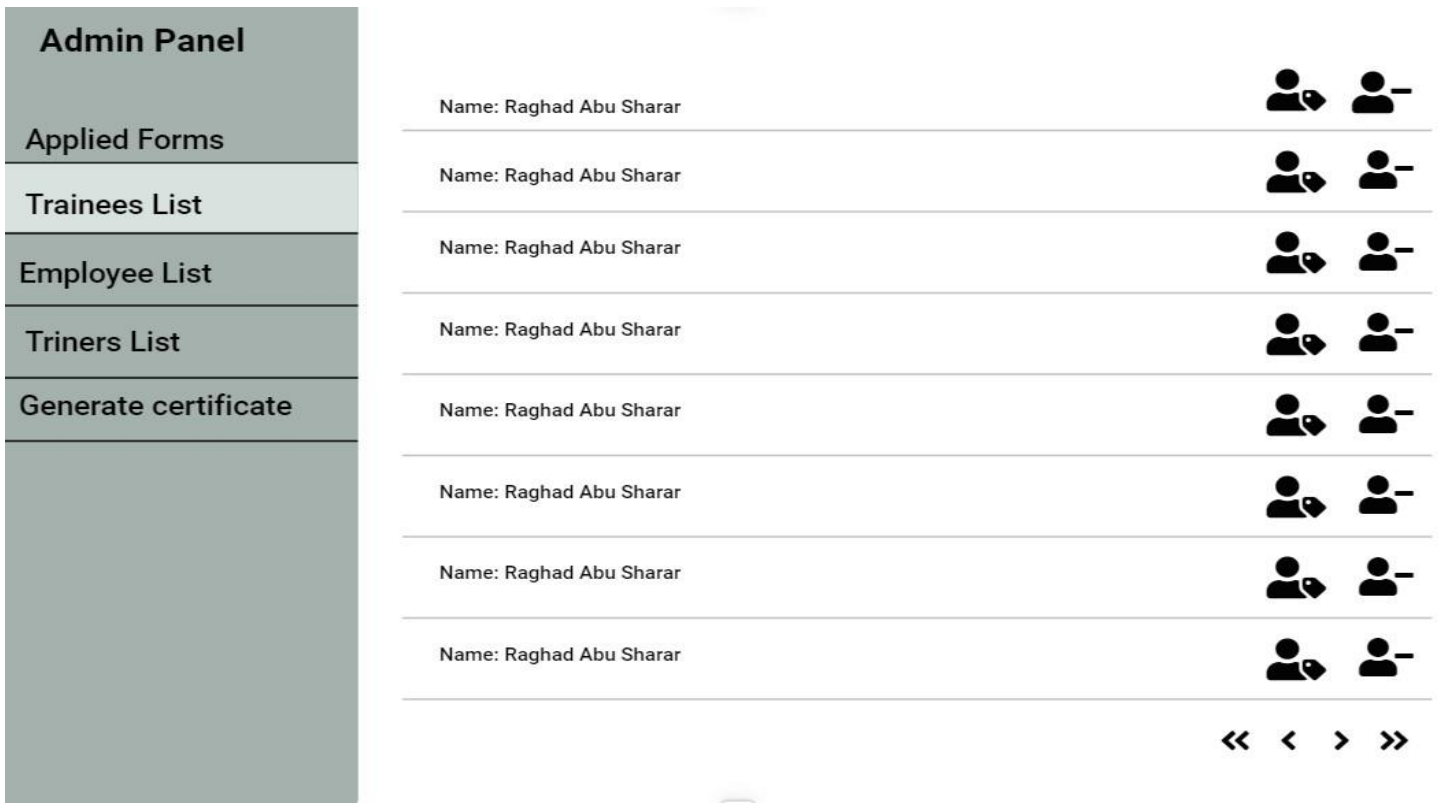


Figure 3.8.1.14 Admin panel: trainees list page

- **Employees list page:** this page shows the company employees with their department, with create an account option to add to trainers list.

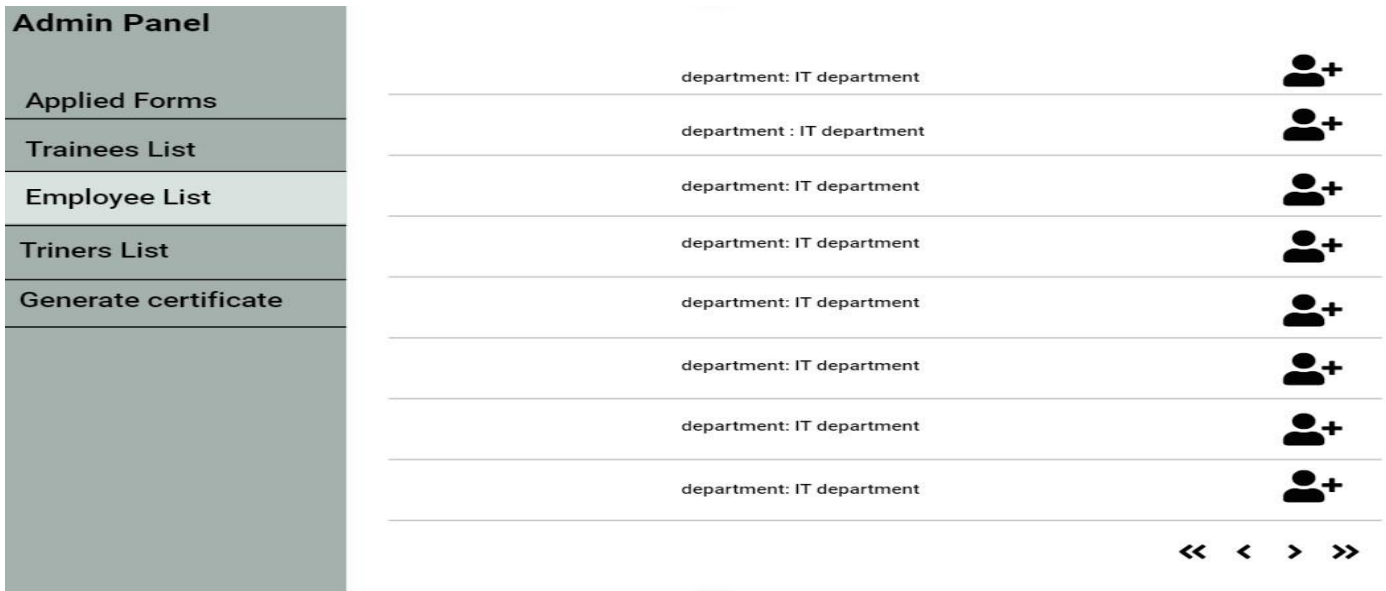


Figure 3.8.1.15 Admin panel: employees list page

3.8.2 Mobile Training Application

First open page:



Figure 3.8.1.16 Training Application: First open page

Trainee profile page:

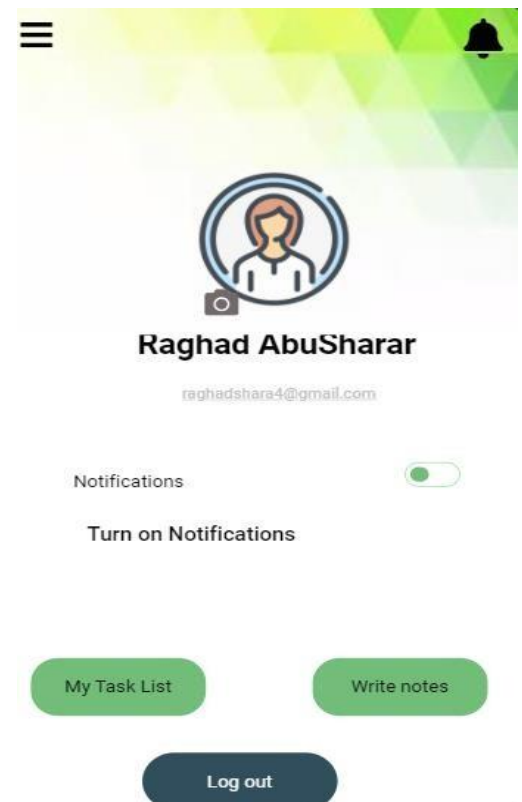


Figure 3.8.1.17 Training Application: Trainee profile

Task list page:

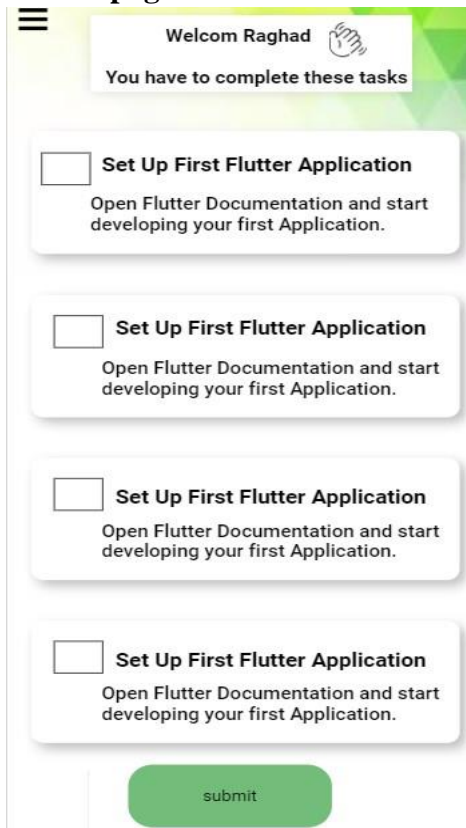


Figure 3.8.1.18 Training Application: Tasks List

Trainee daily report page:

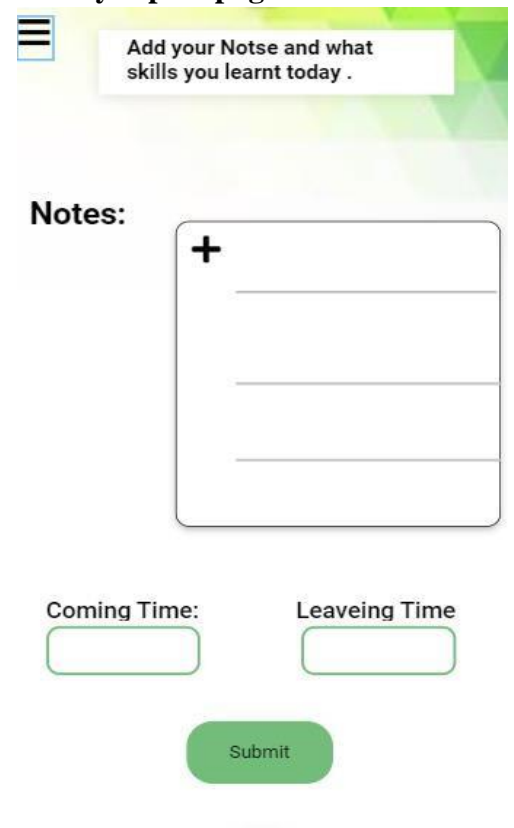


Figure 3.8.1.19 Training Application: trainee report

Trainer profile page:

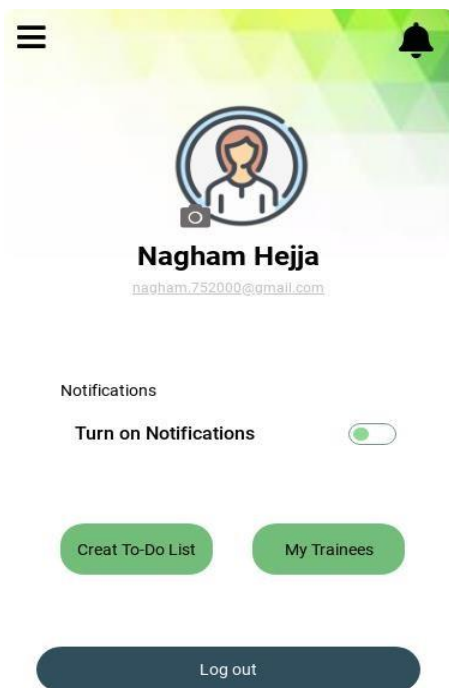


Figure 3.8.1.20 Training Application: Trainer profile

Create task list page:



Figure 3.8.1.21 Training Application: Trainees task list

Trainee evaluation page:

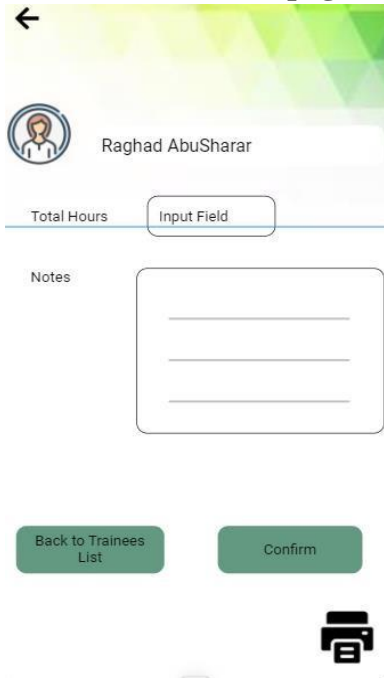


Figure 3.8.1.22 Training Application: Evaluation page

University supervisor profile page:

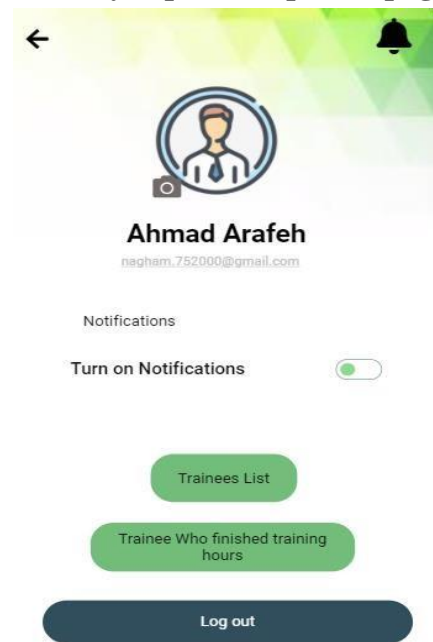


Figure 3.8.1.23 Training Application: Supervisor profile

Trainee access screen page:



Figure 3.8.1.24 Training Application: Trainee access screen

Chapter 4

Software demonstration

- **Introduction**
- **Software needed for development stage**
- **Scope of the system**
- **System programming**
- **System interfaces**

4.1 Introduction:

In this chapter, we will discuss how the system was built. The system implementation stage is one of the most important stages, through which the transition from the theoretical stage, which is the preparatory stage of the system to the practical stage, and then start programming and building the system. We will learn about the tools and programs necessary to develop the system and operate it fully and effectively, and the software that was used.

4.2 Software needed for the development stage:



Figure 4.2.25 Software used in development

We first start writing down the project documentation using Google Tools (Google Docs, Google Slides, and Google Drive) then the diagrams that are attached to the documentation depending on the highest Software Concepts were created with Draw.io. In developing the application, angular Material were used as a Frontend and PhpMyAdmin for storing the created tables in MySQL, and NodeJS as a Backend. Bootswatch and Bootstrap were needed in our Frontend code. Tasks were divided among the group members on Jira Software, and Scrum meetings were held with Jawwal Company during the developing period. The code was written using Visual Studio Code (VS Code). Postman was used for testing the required functions.

4.3 System programming:

The project contains two parts as the follows:

1. **Mobile application:** which is used by trainees to track their task list and record daily training hours. And trainees manage their trainers and create tasks lists. Also, the university training supervisor for tracking trainee's performance.
2. **Web application** (admin panel): which is used by the company for adding and deleting users and updating their information. Also for deactivating their accounts.

4.3.1 The mobile application:

We have relied on the MVVM structure in building the system, as this structure divides the system into three main parts.

They deal with each other with a specific mechanism to achieve the required operations and represent the system relationships through the graphics that explain the functioning of the system and how it relates to the external environment, which facilitates the process of clearly understanding the system. The main idea in this hierarchy is the model view layer, which provides data to the interfaces. This data is used for Creating a user interface.

4.3.1.1 MVVM mainly contains the following levels:

1. Model layer:

This section contains user interfaces that respond to changes in the model view and transmit events performed by the user to model view.

2. View Model layer:

The View Model interacts with the model (data layer), and the View Model can observe it with the View.

View Model can selectively provide hooks for views to pass events to the model.

An important implementation strategy for this layer is to separate the Model from the View, which is a View Model that should not be aware of who is interacting with the show.

3. Layer view:

The role of the viewer in this mode is to observe (or participate) in the Mode View, monitor the change of data, to obtain data to update UI elements.

4.3.1.2 Why using MVVM pattern:

view model acts as an adapter for the model classes and gives the ability to avoid making any major changes to the model code. Also, developers can create unit tests for the view model and the model, without using the view. The unit tests for the view model can exercise exactly the same functionality as used by the view.

4.3.1.3 MVVM model

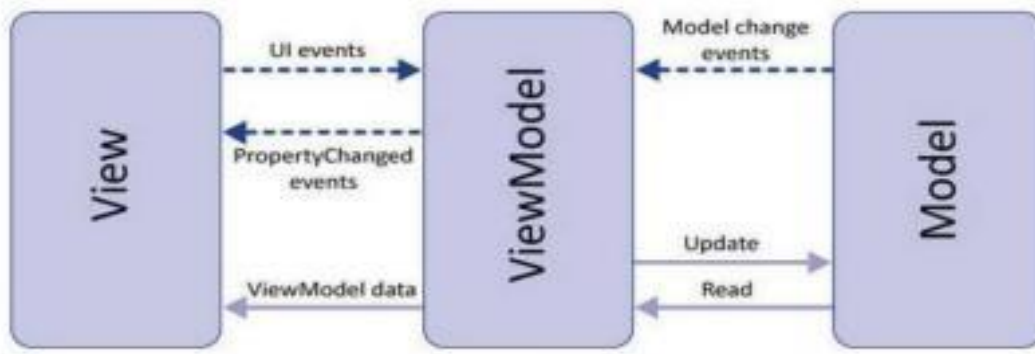


Figure 4.3.1.3.26 MVVM model

4.4 The web application:

4.4.1 Backend Node.js:

It is an open-source development platform for executing JavaScript code server-side.

4.4.2 PhpMyAdmin:

It is an open-source tool to enable system administrators to manage MySQL databases on the Internet. it create/drop/modify /delete tables and fields.

Chapter 5

Testing

- **Introduction**
- **Functional requirements**
- **Non-functional requirements**
- **Domain requirements**

5.1 Introduction:

In the stage of testing the system, we make sure that the system works correctly without any problems, and we also make sure that the functional and non-functional requirements of the project are completed, and that the system works with accuracy and high speed in completing tasks and displaying information. The stage of testing comes after the design and implementation of the system.

5.2 Validation:

All information entered in all fields in the application are tested to ensure that the data entered by the user matches all conditions as follows:

- Customize the field in proportion to the entry.
- The process will not be executed if wrong data is entered.
- The process will not be executed if the users entered number of characters exceeded the permissible limit.
- Ensure that there are actual users in the database.

5.3 Screen testing:

We applied the screen testing on all screens in the system. Each screen was tested on the chrome web browser and we tested the response of each screen for all screen sizes.,

5.4 API request testing:

The system units were fully tested and ensured how they worked. The result of the examination was successful. The following tables review the testing we have done:

Table 5.4.26 User sign-up

#	case	Input	Expected output	obtain output	Pass/Fail
1	Correct information	Email=181120@ppu.edu.ps Password=12345678	Successful Signup	Okay	Pass
2	Email already exist	Email= 181120@ppu.edu.ps	Error message	False	Fail
3	Password does not matching system validation (contains at least 8 characters)	Password= 12345678	Prevent posting the request	Expected	Fail

Table 5.4.27 User log-in

#	case	Input	Expected output	obtained output	Pass/Fail
1	Correct information	Email=181120@ppu.edu.ps Password=12345678	Successful login	Okay	Pass
2	Invalid Email	Email= 181120ppu.edu.ps	Error message	False	Fail
3	Password does not matching system validation(contains at least 8 characters	Password= 12345678	Prevent posting the request	Expected	Fail
4	Invalid password (should contain numbers and letters)	naghaam	Error message	Expected	Fail

5.4.1 Log-in validations:

An alert message is shown when logging in without filling the required fields.

Email message will be shown when entering invalid email

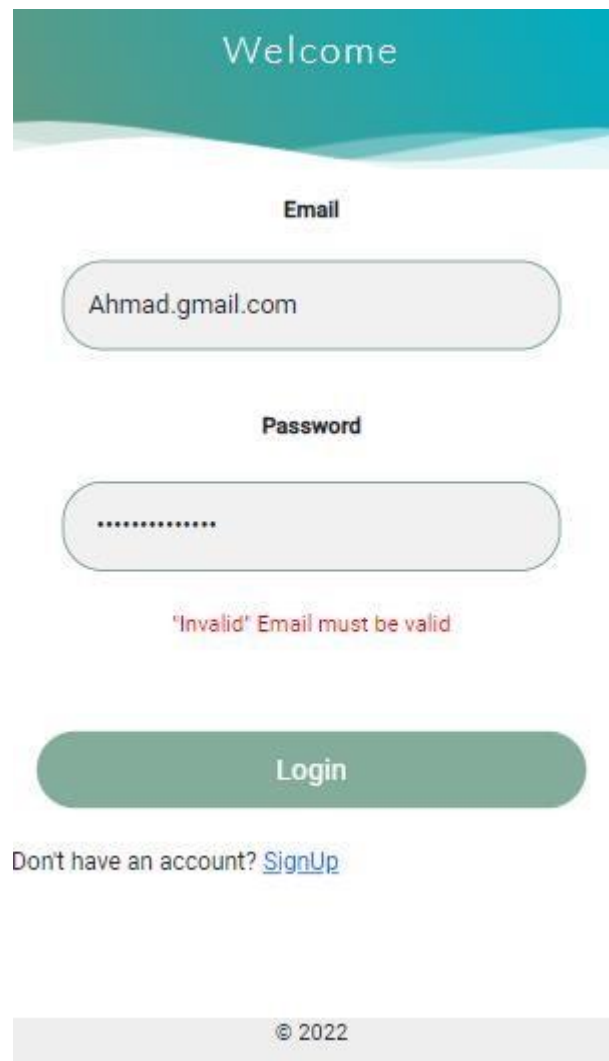
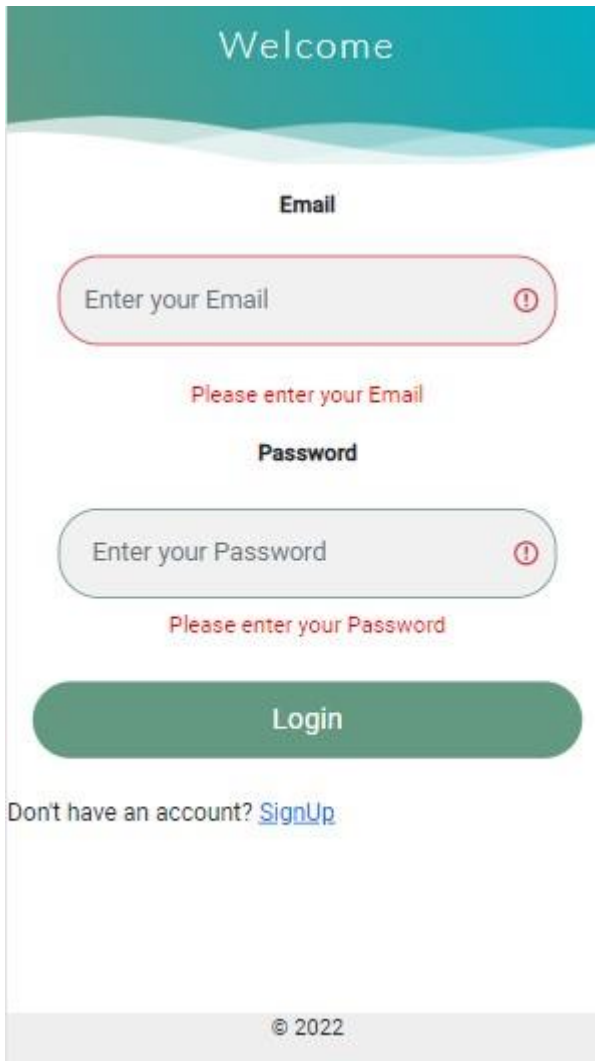
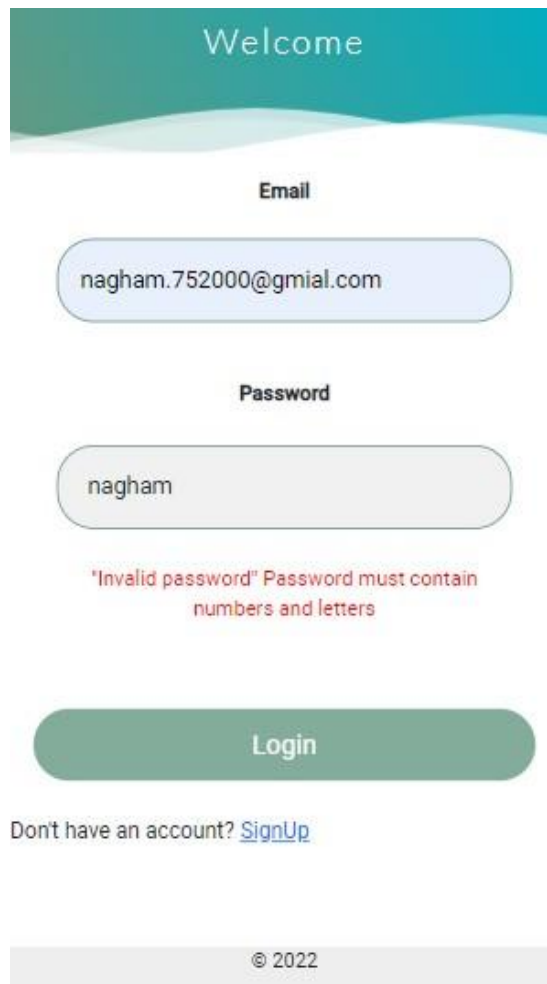


Figure 5.4.1.27 Login validations (required fields)

Figure 5.4.1.28 Login validations(invalid email)

Invalid password message will be shown when entering a password that does not contain numbers.



The image shows a login form with a teal header containing the word "Welcome". Below the header, there are two input fields: "Email" and "Password". The "Email" field contains the text "nagham.752000@gmial.com". The "Password" field contains the text "nagham". Below the password field, there is a red error message that reads: "Invalid password" Password must contain numbers and letters. Below the error message is a green "Login" button. At the bottom of the form, there is a link that says "Don't have an account? [SignUp](#)". At the very bottom, there is a footer that says "© 2022".

Figure 5.4.1.29 Login validations (invalid password)

Chapter 6

Conclusion

- **Recommendations**
- **Future work**
- **References**

6.1 Recommendations:

In the view of the technological development that our current era is going through, we recommend students, universities and companies who need to get in the training process, to take advantage of this system because of the ease in dealing with it, it also facilitates the training process for all parties and makes them have an easier and more simple training than before.

We recommend working on implementing the system at the university and the company for at least one semester, noting the problems that result from using the system and working to solve them.

6.2 Future work:

In the future, we are looking forward to adding many features to the current system such as:

- Adding a live chatting feature that facilitates the communication between the trainer and the trainee.
- Adding a notification feature whenever the trainer adds a task list for the trainee and when a specific trainee finishes his training hours.
- Making a task list for each student individually according to his interests and skills.
- Provide an authentication code before continuing filling out the form.
- Expand the system to cover a large number of companies.
- Add sign up using Social Media accounts (SSO).
- Log in using one time authentication code sent by phone number.

6.3 References:

6.3.1 Sites used in developing the system:

<https://material.angular.io/> <https://www.npmjs.com/>

<https://www.mongodb.com/docs/drivers/node/current/fundamentals/crud>

[/ https://bootswatch.com/](https://bootswatch.com/)