Palestine Technical University-Kadoorie

Faculty of Engineering and Technology

Computer Systems Engineering Department

"A Website for the Palestinian Tabou"

Prepared By:

Fayha' AbuSalah – 201910068

Mays Hawwa – 201910360

Mohammad Eneirat – 201911370

Osama Marei – 201911195

Supervised by:

Dr. Yousef Daraghma.

Tulkarm, Palestine
June, 2023

This document was submitted in fulfilment of the requirements for the " Graduation Project" course, during the academic year 2022 - 2023 (Second Semester).

ACKNOWLEDGEMENT

First of all, we would like to thank God Almighty for giving us the opportunity and guidance to achieving our goal and to be successful in this part. Then we would like to thank our families for their upbringing of and their tireless efforts and support in every path we take to achieve our dreams and goals.

We would like to express our special thanks to mentor Dr. Yousef Daraghma for his time and efforts he provided throughout the year. Your useful advice and suggestions were really helpful to us during the project's completion. In this aspect, we are eternally grateful to you.

Finally, all thanks to group members for sharing the positivity and invaluable assistance.

ABSTRACT

The idea of this project is designing the TABOU website to meet the needs of the Palestinian citizen.

During our visit to the Land Registry Directorate in Tulkarm, we noticed that the completion of transactions by citizens or employees, from selling to buying, mortgaging, sorting, and many others, require time and burden, in addition to the need for a large number of official papers attested by departments and other official bodies. Therefore, we design a website for the Tabou that enables users to log in, create transactions for selling or buying lands, and track their transactions. The website is designed according to the requirements collected from the Land Registry Directorate.

We aim to operate a web-based website - for the Directorate of Land Registry - that facilitates citizens and employees, so that transactions are completed easily to the fullest without the need to waste time.

TABLE OF CONTENTS

LIST OF FIGURES

List Of TABLES

CHAPTER 1
INTRODUCTION

OVERVIEW

We are now in a time when dependence on the Internet fully has become in where you can do whatever you want while you are in your home or workplace without the slightest personal effort. Hence, the idea of computerizing a web system to facilitate land transactions.

At the present time, there is an increasing demand for trade in lands, buying and selling, etc. from private transactions in lands.

If a citizen wants to complete a transaction related to land he owns, he first goes to the Land Registry Department to inquire about the requirements of this transaction, and to bring it from the official departments. If there is a defect in one of the papers, he must visit the department again, and then return to the Land Registry Department, etc.

But this is costly at the expense of personal time and material; Because there is congestion in the department...etc.

From here, our social role emerged in creating what serves Citizens and facilitates this process for them in a smooth manner with the least possible effort and fatigue by creating a website to computerize a system for land transactions such as buying, selling, etc. of various land transactions and providing them with immediate updates on their transactions. Which they provided with high quality and accuracy “keep it simple, keep it smooth”.

Problem Statement

As we mentioned briefly in the introduction about several issues in the subject that would waste the citizen's time in order to complete a transaction.

As the completion of a transaction on your land needs to take a vacation from your work and personally go to the Land Registry and complete the transaction, and the problem lies in whether this transaction is not done through one government department and you need several other government departments, for example (a sale contract from the court, clearance of the municipality, etc.)

The matter will become more difficult and complicated, which would exhaust the citizen and assign him financially and physically instead of the delay that will occur when waiting for the roles, and if you are outside the borders of the country and need to do one of the land transactions such as selling, for example, you will have to authorize someone to follow these procedures and you will not be able to complete them in person.

At this stage, we are planning to complete a site through which you can complete transactions electronically as much as possible as much as possible by providing a smooth and simple user interface that allows the user to open a transaction such as selling, for example, and track it electronically and find out if there is a defect in this transaction immediately and know the defect and work to fix it as soon as possible to complete the procedures faster and less tired and effort.

OBJECTIVE

1. Building a website that keeps a record of transactions and citizens' rights and properties.
2. Ensure that the website is responsive and easy to handle.
3. Ensure data protection and preservation from damage and loss instead of papers.
4. Ensure that the site achieves its desired goal, which is to make it easier for citizens, save money and time, and reduce the burden.
5. The platform should be easy to navigate and use, even for individuals with limited technical knowledge or experience

CHAPTER 2

RELATED WORKS

During the research period that we conducted, we reviewed studies and research related to the subject that serve the same goal of our website, and we also visited the Land Registry Department in Tulkarm Governorate more than once to obtain sufficient information.

2.1 Jordanian Department of Land and Surveys Website (DLS)

The Department of Land and Survey is a governmental institution responsible for land management, surveying, and mapping in Jordan. The website serves as a platform to provide information and services related to land and surveying activities in the country.

By visiting the website, you may find a range of information and resources, including:

1- Land Registration: The DLS likely provides services related to land registration, title deeds, and property ownership.

2- Surveying and Mapping: The website may contain information about surveying and mapping services offered by the DLS.

3- Legal and Regulatory Framework: You might find information about land-related laws, regulations, and policies implemented by the DLS and the government of Jordan. [2.1]

Figure 1 DLS.Jo Website

2.2 The Digital Portal for Government Services (Hukoomi)

The digital portal for government services is a government digital platform that allows citizens to submit requests for government transactions in an easy and convenient way. The Ministry of Communications and Information Technology works to make all government transactions available in cooperation and coordination with government ministries and departments. The ministry also takes into account Safe, fast and integrated.

The portal provides an integrated guide to inform users of the information related to each government service, in terms of the conditions and requirements for accepting applications, the time required for its completion, the value of the required fees, and other information.

You can also download the "Hukoomi" smart devices application, which represents an additional channel for the digital portal services, whereby the application services can be enjoyed through the smart devices of the Android system as well as the IOS system. [2.2]

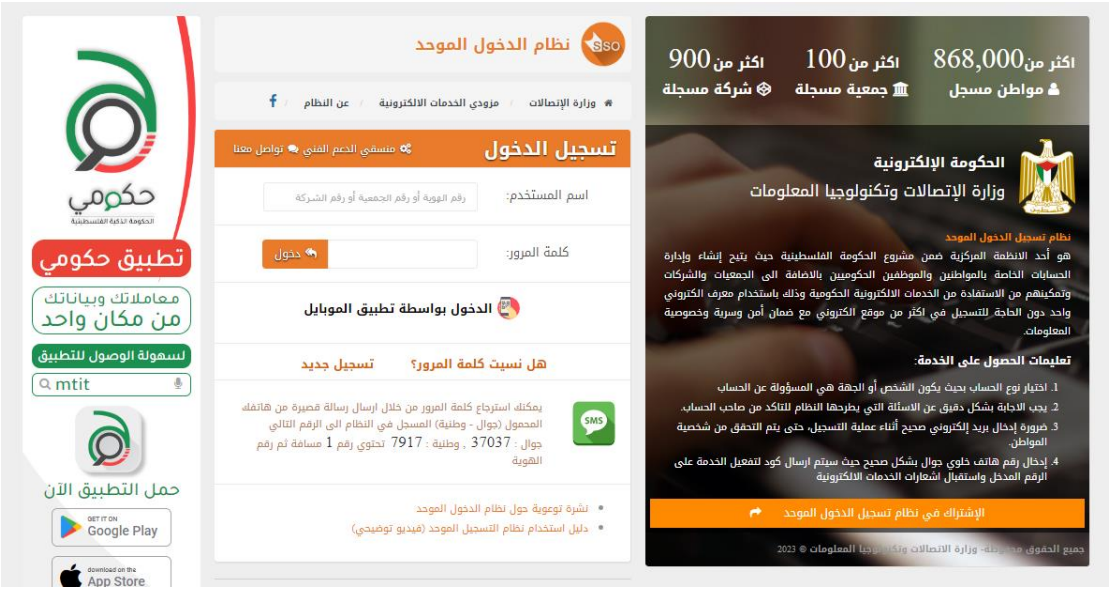


Figure 2 Hukoomi Website

CHAPTER 3
METHODOLOGY

The methodology in this project follows software development methodology. We start by requirement analysis, designing the system, implementation and testing.

3.1 Functional Requirement

Example of The Organizational Structure (Land Registry):

Table 1: Structure of Palestinian Tabou Website

Position Name		Privilege
Admin	**مدير الدائرة**	**1**
Employee	**الموظف**	**2**
Citizen "User"	**المواطن**	**3**

1- User:

The system must provide the Citizen " User " with the ability to:

- Logging in to his account.
- Logging out of his account.
- Submitting of a land sale transaction.
- Submitting an Limiting a legacy transaction.
- Submitting a mortgage transaction.
- Submitting a land fragmentation transaction.
- Submitting a land sorting transaction.
- Know the status of the transaction submitted instantly.
- Inquire about his lands.

2- Employee:

The system must provide the employee with the ability to:

- Creating accounts for users – citizens.
- Modifying user data – citizens.

- Receiving and processing user-citizen transactions such as buying and selling, etc.
- Pay the transaction cost to complete it.
- Inquiry from the municipality about the clearance of the user - the citizen, the description of the land and the land plan.
- Inquiry from the Ministry of Interior about the personal data of the citizen.
- Inquiry from the Survey Department about the survey plan.
- Inquiry from the Sharia courts about Limiting a legacy and the sale contract.
- Inquiry from the Ministry of Foreign Affairs about limiting a legacy document.
- Receiving and processing the sale and purchase of land.
- Receiving and processing the land sorting transaction.
- Receiving and processing land fragmentation transactions.
- Receiving and processing a land mortgage transaction.
- Receiving and processing an Limiting a legacy transaction.

3- Admin:

The system should provide the manager with the ability to:

- Creating accounts for employees.
- Inquiry from the municipality about the clearance of the user - the citizen, the description of the land and the land plan.
- Inquiry from the Ministry of Interior about the personal data of the citizen.
- Inquiry from the Survey Department about the survey plan.
- Inquiry from the Sharia courts about Limiting a legacy and the sale contract.
- Inquiry from the Ministry of Foreign Affairs about Limiting a legacy document.
- Assign leave to an employee and freeze his account.
- Tracking employee's workflow.

3.3 Non-functional requirements:

Non-functional requirements are constraints on the operation of the system. This section discusses; usability requirements, security requirements, accuracy requirements and reliability requirements.

1- Security Requirements:

Requirements that describe how access to the application will be controlled and how data will be protected during storage and transmission.

- Each user has a password-protected account so no one can access using his account without the correct password, and using the for the user.
- The password for the user is encrypted in the database, so if someone accessed the database, he can't decrypt; we do that using bcrypt library, this library take the password and encrypt it 4 times; hash password = bcrypt. hash (password, saltround).
- Any Form data will be checked if it is valid and the system is not vulnerable to attacks such as SQL injection.
- Every end-point secure by token and bearer-token, the token encrypted by JWT; as token = JWT.sign (payload, key); encrept done by use key, this key defined by developer.

2- Usability Requirement:

Operational characteristics related to users.

- The user will interact with the application via a web page programmed using HTML and other web-related technologies.
- The interface is user-friendly, easy to learn and easy to use.
- Admin side can inquire about all transactions, and he can inquire about any transaction by search

3- Reliability Requirement:

We are planning to make the system as much reliable and dependable as much as possible.

- The system will be available 24/7, which is accomplished by hosting the webpage on a server, but the system processes a limited number of transactions.

4- Accuracy:

- Our system will ensure correct output is produced; When a citizen raises a transaction, it will reach a specific employee in the department to audit it, and answer with rejection or acceptance based on what has been submitted and the audit is done well; If a defect occurs in one of the transactions, the cause of the problem is displayed in detail to the citizen on his account through a text message when inquiring about this transaction.

3.4 Process Model

We know that the project will go through a lot of changes in both functionality and design throughout the development time, and these changes happen due to three reasons, stated below:

1. We know what the general shape of the project is and what we want it to do, but we haven't decided on the details which will require some changes in the design in the future.

2. Because most of us will be learning new frameworks for programming various parts of the project (i.e., front-end, back-end, etc...), So with time, we'll get more familiar with these frameworks and therefore, find better and more efficient ways to solve problems and design a different approach for tackling these problems.

3. Our desire to add a new feature or to remove one if found necessary.

So, we wanted a development model that is flexible and easy to integrate changes with, and because of that, we decided to use an agile model which compared to other development models offers great advantages, to list some:

❖ Agile is an incremental approach and each increment is delivered within a certain time interval and between these time intervals comes the opportunity to apply changes to the design, functionality, or logic if needed, unlike the waterfall model in which after the development starts you can't change the design.

❖ The way that agile works, it gives us and the client a working project within some time interval, and that project can be tested for bugs and given to people to use and give feedback and change some aspects in the next time interval, unlike other models that don't provide a fully working product until the end of the development phase.

❖ Agile divides the project into increments which makes it easier to design these increments, rather than designing a complex system in one phase.

❖ Agile provides different models each one provides different techniques and tools to apply and achieve agile model principles, but in our case, we used the Scrum model, for the following reasons:

- It is easy to use, mainly because it provides you with general guidelines to follow and it is up to you how to do them, which is easier than learning a new workflow.

- Fast response to changes, where a new change can be easily added to the backlog, and therefore taken into consideration in the future plans.

For our project to be done, we plan to follow the Agile process model. Agile process model refers to a software development approach based on iterative development. We are planning to break project tasks into small iterations (scrums), where each scrum will take two weeks. The division of the entire project into scrums helps to minimize the project risk and to reduce the overall project delivery time requirements.

Figure 3: Agile Model LifeCycle

CHAPTER 4

SOFTWARE DESIGN

4.1 Use Case Diagrams

Use cases are typically documented in a use case diagram or written as textual descriptions. They serve as a means of capturing and communicating the functional requirements of a system, helping to define its behavior from the user's perspective. Use cases are often used in the early stages of software development to clarify and validate system requirements, and they continue to be useful throughout the development lifecycle.

4.1.1 User Use Case Diagram

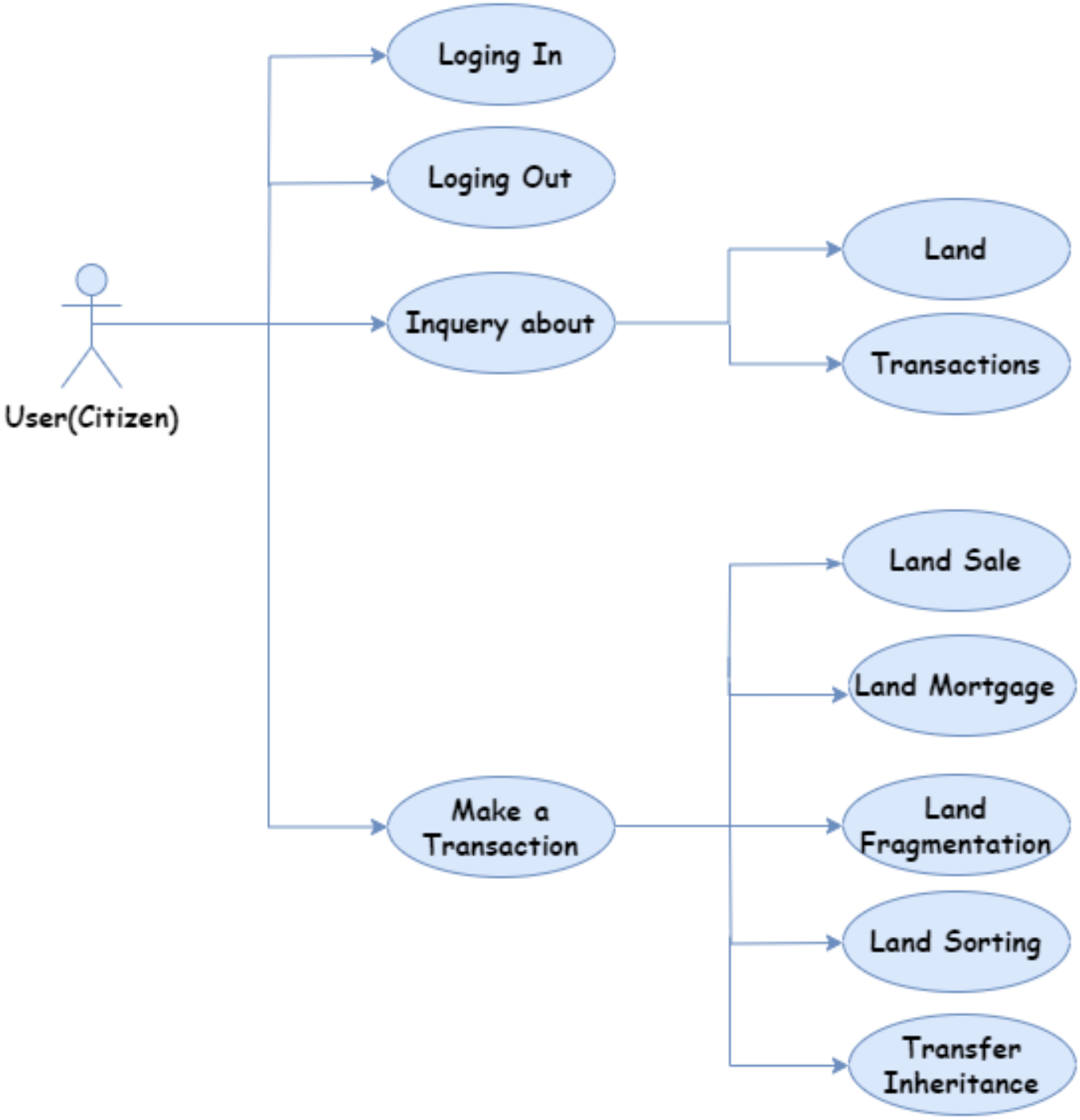


Figure 4: User's Use Case Diagram

4.1.2 Employee Use Case Diagram

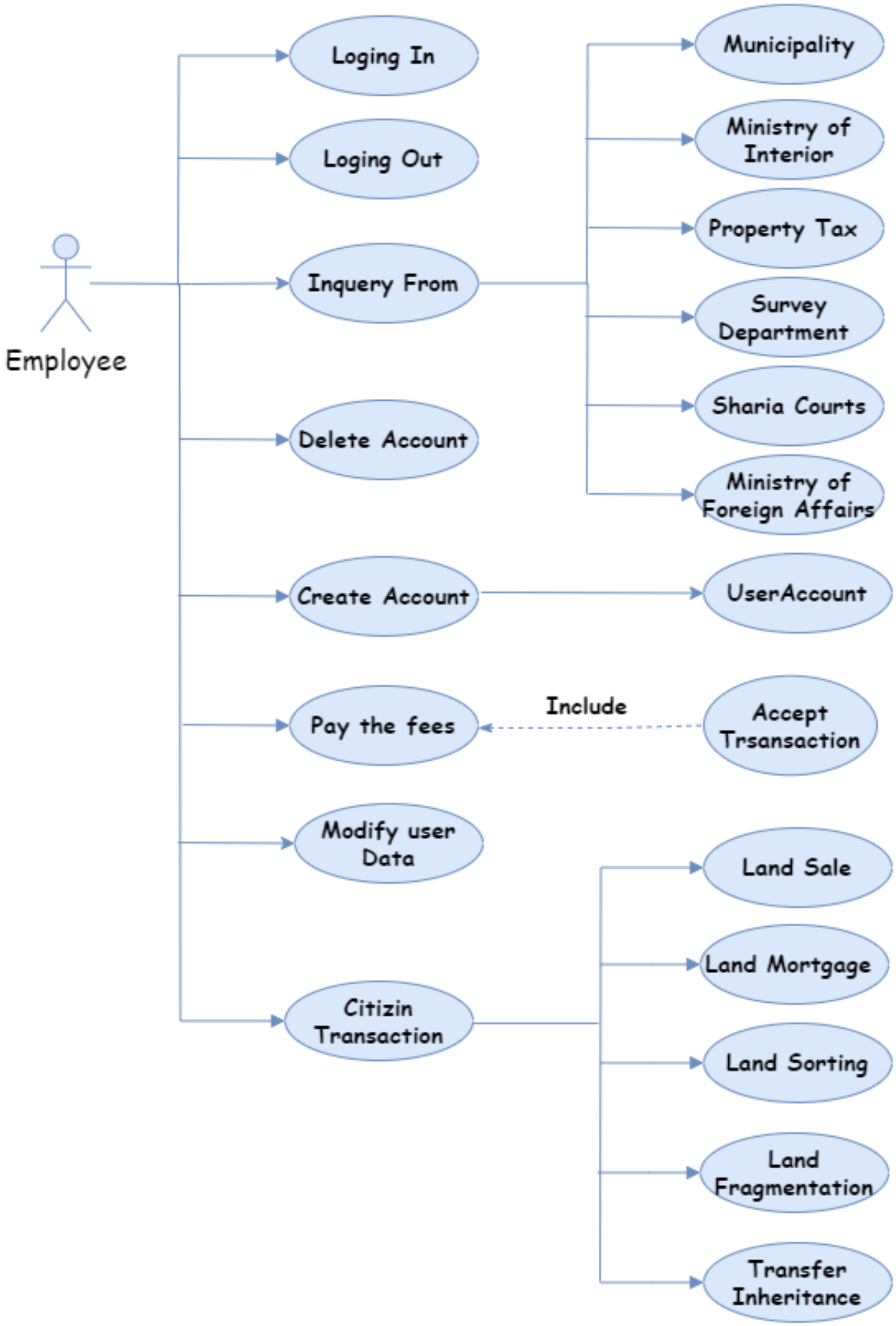


Figure 5: Employee's Use Case Diagram

4.1.3 Admin Use Case Diagram

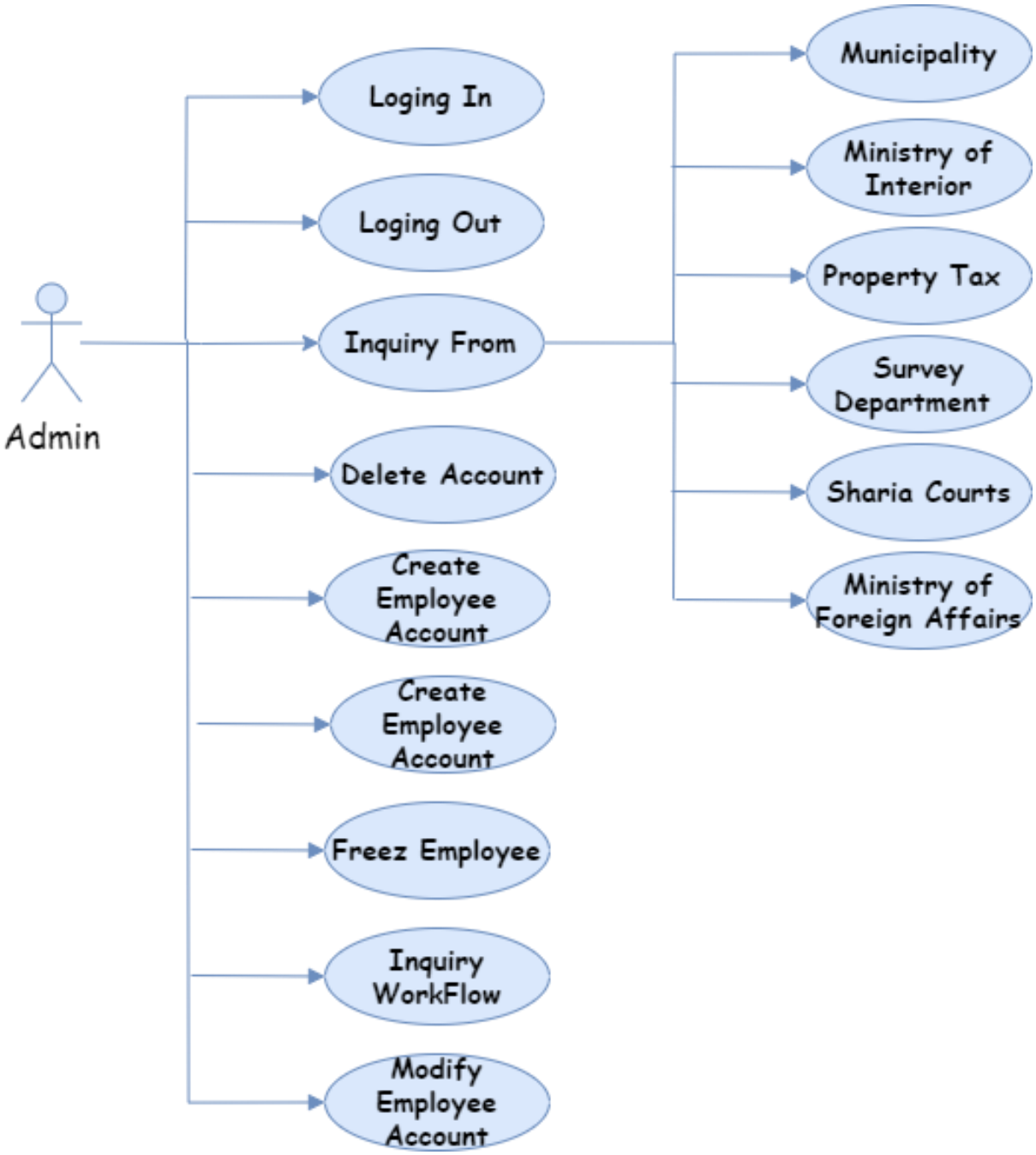


Figure 6: Admin's Use Case Diagram

4.1.4 Employee Use Case Diagram Description

Employee's Use Case Description 2 Table

Use Case	Description
Inquiry from Several Government Circles:	We inquire from different government departments about some of the papers attached by the citizen to ensure the validity of his information and approval of the transaction to carry out its implementation.
Create Account:	The employee can create an account for a citizen after the citizen submits a request to create an account, taking into account the inability to create more than one account for one citizen if his account is active.
Modify User Data:	The employee can modify the citizen's account data after a request from the citizen to amend. For example, modifying the phone number or address...etc.
Citizen Transaction:	The employee receives all the various citizen transactions, selling, sorting, retailing, mortgaging and transferring inheritance. Where the employee deals with these transactions by acceptance or rejection after verifying all the attached data from the various departments, then the transaction is accepted or rejected with an explanation of the reason for rejection.

4.1.5 User Use Case Diagram Description

Table 3 User Use Case Description

Use Case	Description
Login:	The citizen must go to the land Registry and bring the required documents such as identity card, etc. to create an account for him in the database with the land Registry, and then give him a password and an email, and then he enters the password and email on the entry page so that the system directs him to his personal account.
Inquire about his land:	The citizen will have to log in to his account and then click on Show Lands, and all private newspapers in his lands will be displayed.
Transaction query:	The citizen will have to log into his personal account and then enter the Transactions Inquiry box, and all the transactions that he has submitted will appear to him and the status of each transaction if it has been completed or if there is a defect that occurred. Do what is required of him to complete this transaction.
Movements:	When clicking on the Transactions option, the citizen will see all the transactions that he can perform, and when clicking on a transaction, a form will appear for him, containing all the details and papers required to successfully complete this transaction. When you click upload, this transaction will be sent with all the details entered to the Land Registry for consideration and response whether it is correct or there is something wrong with it.

4.1.6 Create User Account Fully Use Case Description

Create User Account full use case Description 4 Table

UseCase Name:	Create User	
Scenario:	The employee wants to create a user account.	
Triggering event:	The employee logging in his account, clicks on "Create New User."	
Brief description:	This use case allows employee to create new accounts to have access to the system.	
Actors:	employee.	
Related use cases:	No related use cases	
Stakeholders:	Employees	
Preconditions:	1. The employee is logged in 2. The new user email and id is be unique.	
Postconditions:	1. Success: The User entered data is stored in the user account, a confirmation is sent to the entered email.	
Flow of activities:	Actor:	System:
	The employee enters the user's username, password, email, address,birthdate, ID and phone number	1. The system prompts the employee for the user's username, password, email, address, ID, birthDate and phone number, The User is created. 2. The system displays a message indicating that the user is created.
Exception conditions:	The user account was not created: The Employee entered invalid user data, or chose to cancel the account creation request. In either case no account will be created.	

4.2 Activity Diagrams

Activity diagrams are graphical representations used in software engineering to depict the flow of activities or processes within a system. They provide a visual representation of the workflow, showing the sequence of actions, decisions, and parallel or concurrent activities that occur within a system.

Activity diagrams are part of the Unified Modeling Language (UML), which is a standardized modeling language commonly used in software development. They are especially useful for modeling the behavior of complex systems, business processes, or use cases.

Activity diagrams provide a high-level overview of the system's behavior, helping to analyze and understand complex processes, identify bottlenecks or inefficiencies, and communicate the system's functionality to stakeholders, including software developers, designers, and business analysts. They are commonly used in requirements analysis, system design, and software documentation.

4.2.1 User Activity Diagram

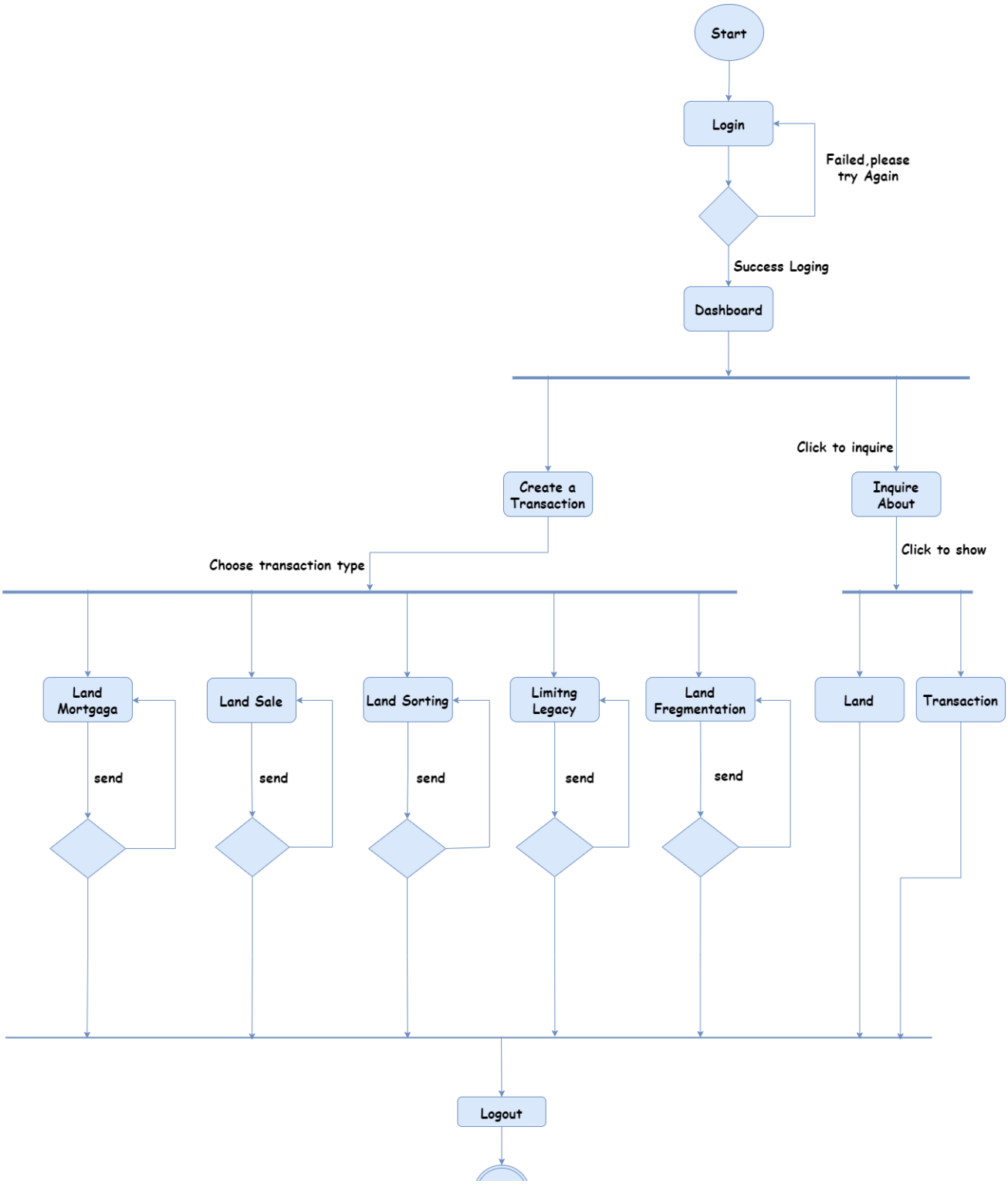


Figure 7 User Activity Diagram

4.2.2 Employee Activity Diagram

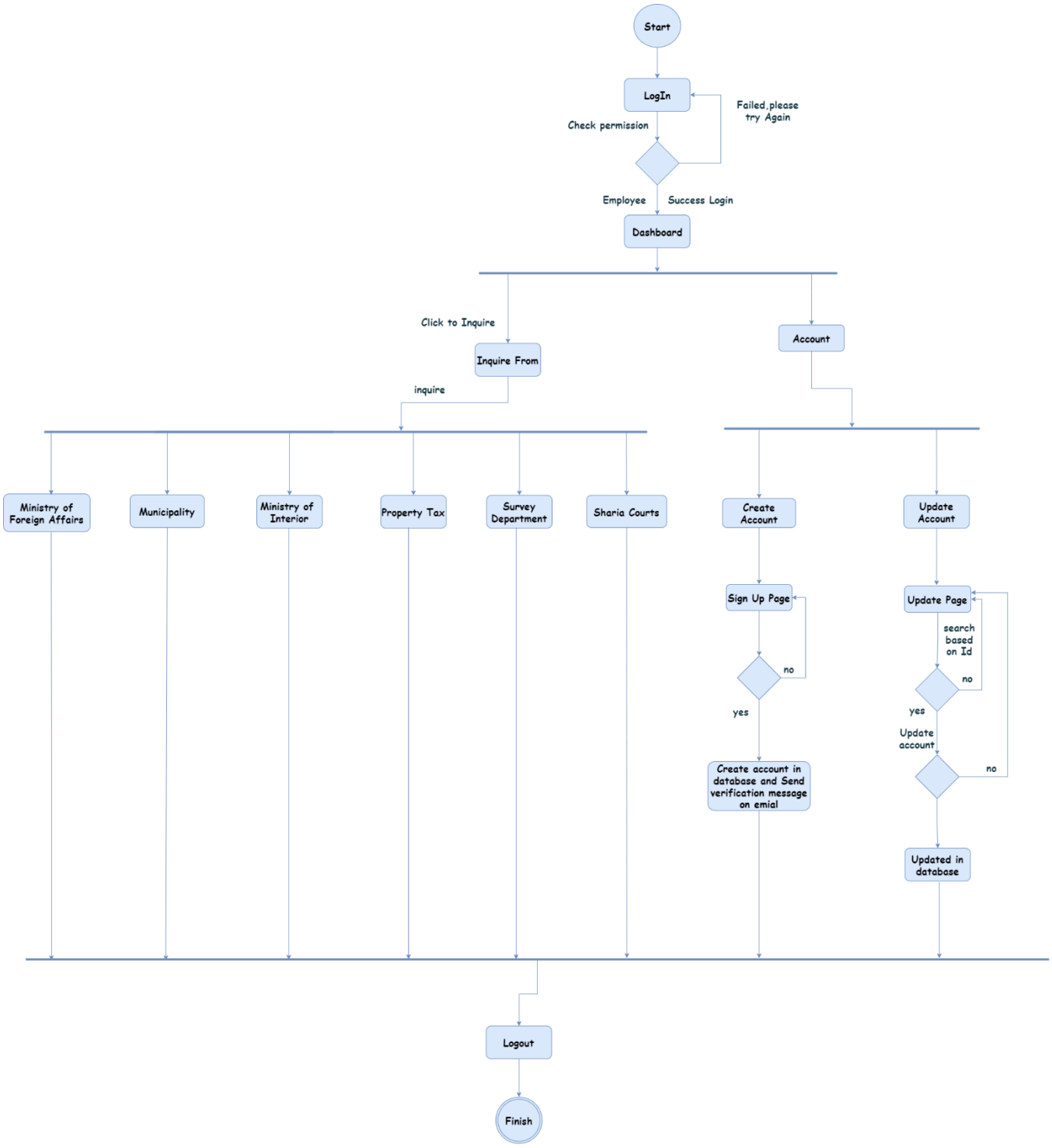


Figure 8 Employee Activity Diagram

4.2.3 Employee Transaction Activity Diagram

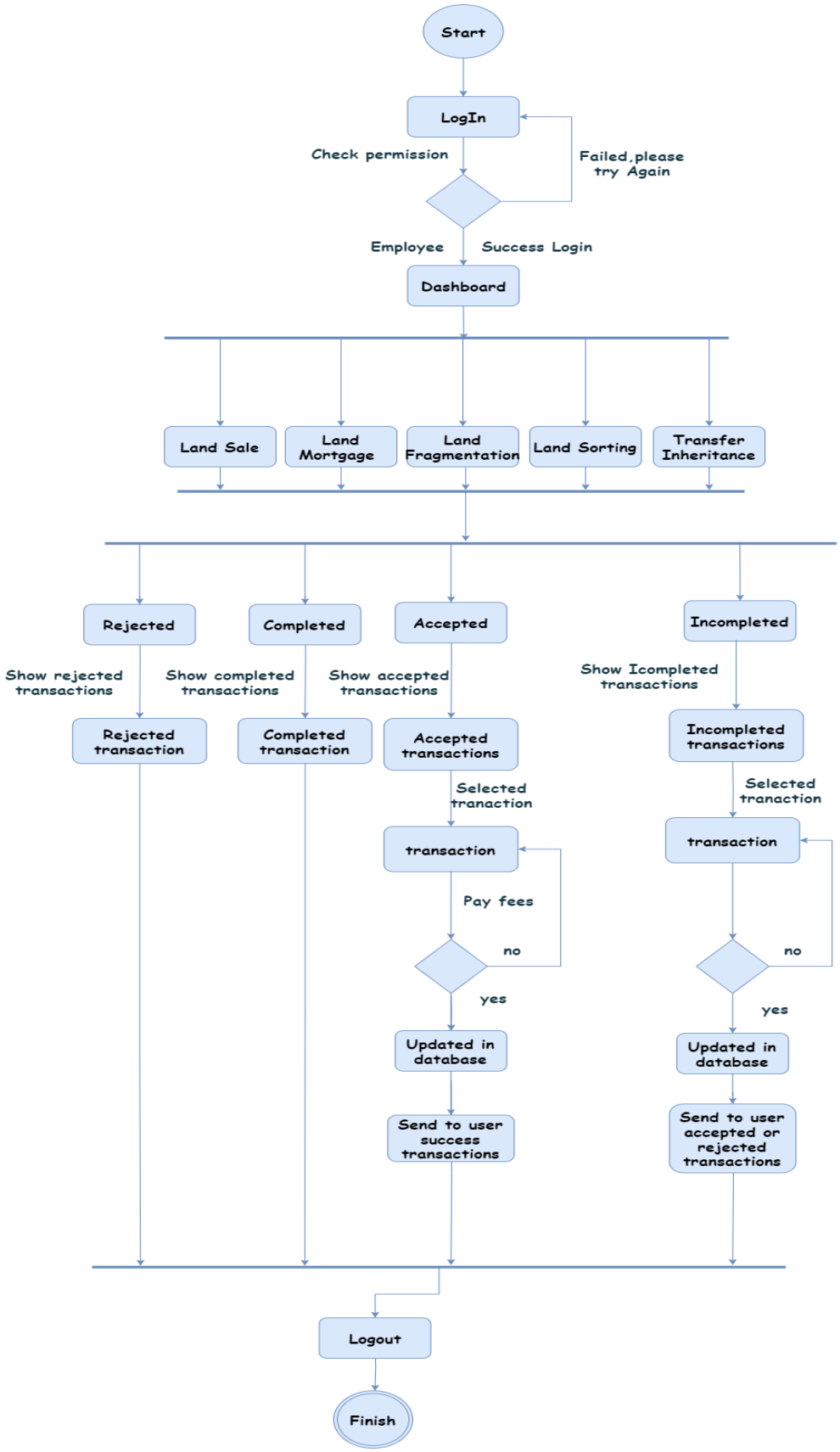


Figure 9 Employee Transaction Activity Diagram

4.2.4 Admin Activity Diagram

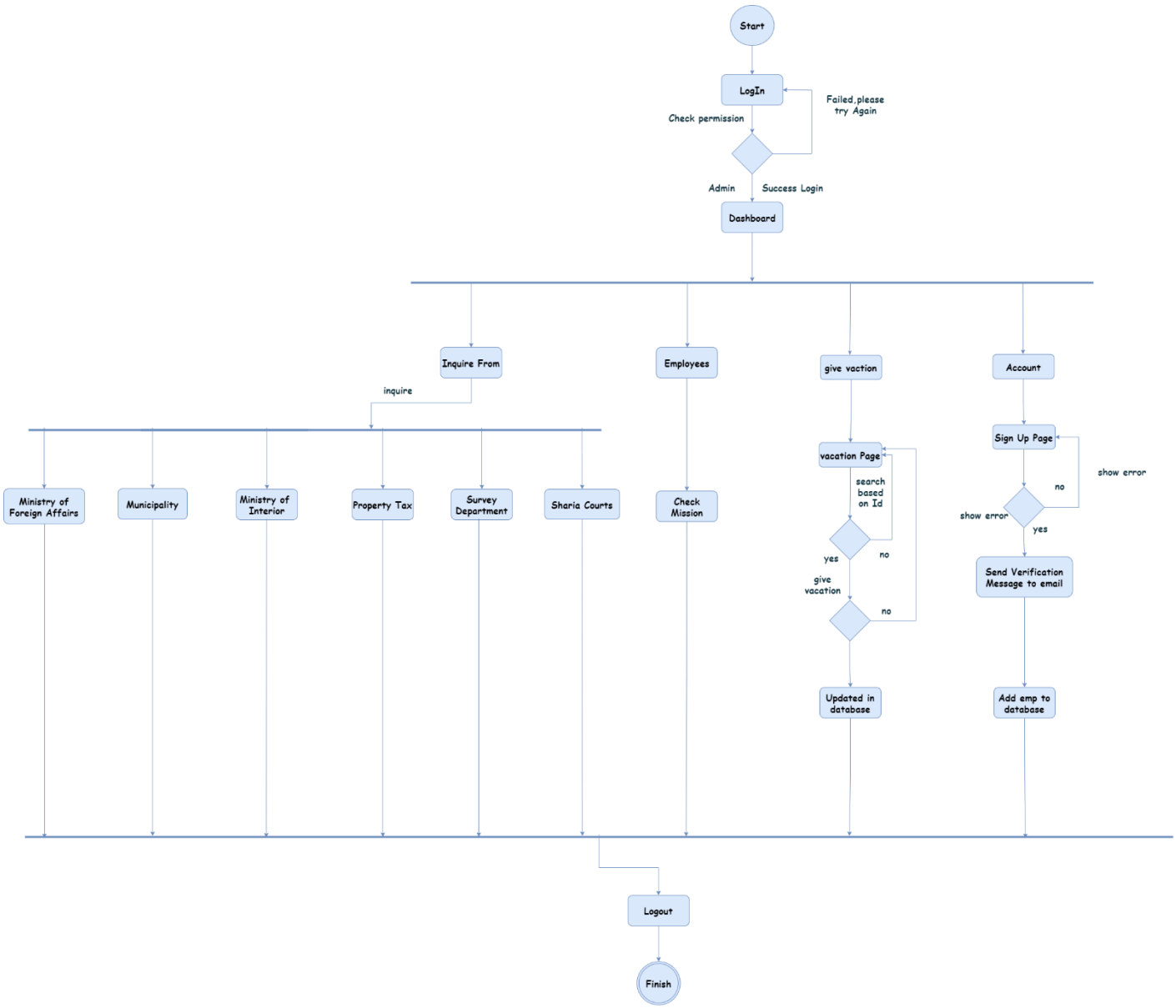


Figure 10 Admin Activity Diagram

4.3 Sequence Diagram

A sequence diagram is a type of UML (Unified Modeling Language) diagram used in software engineering to visualize the interactions and order of messages between different objects or components within a system. It depicts the dynamic behavior of the system, showcasing the sequence of events or method calls that occur over a specific period.

4.3.1 Log-in Sequence Diagram

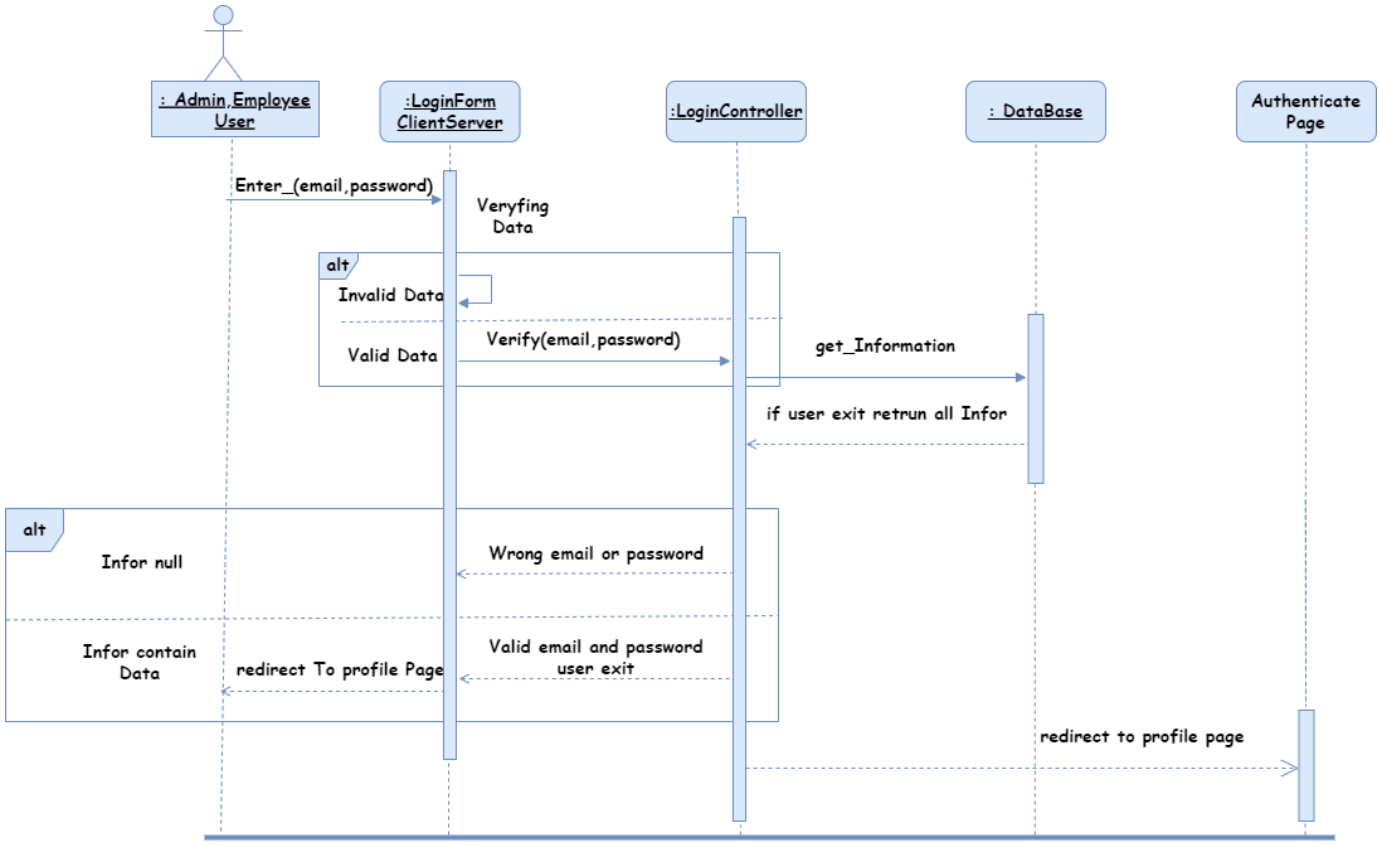


Figure 11 Log-in Sequence Diagram

4.3.2 Transaction Sequence Diagram

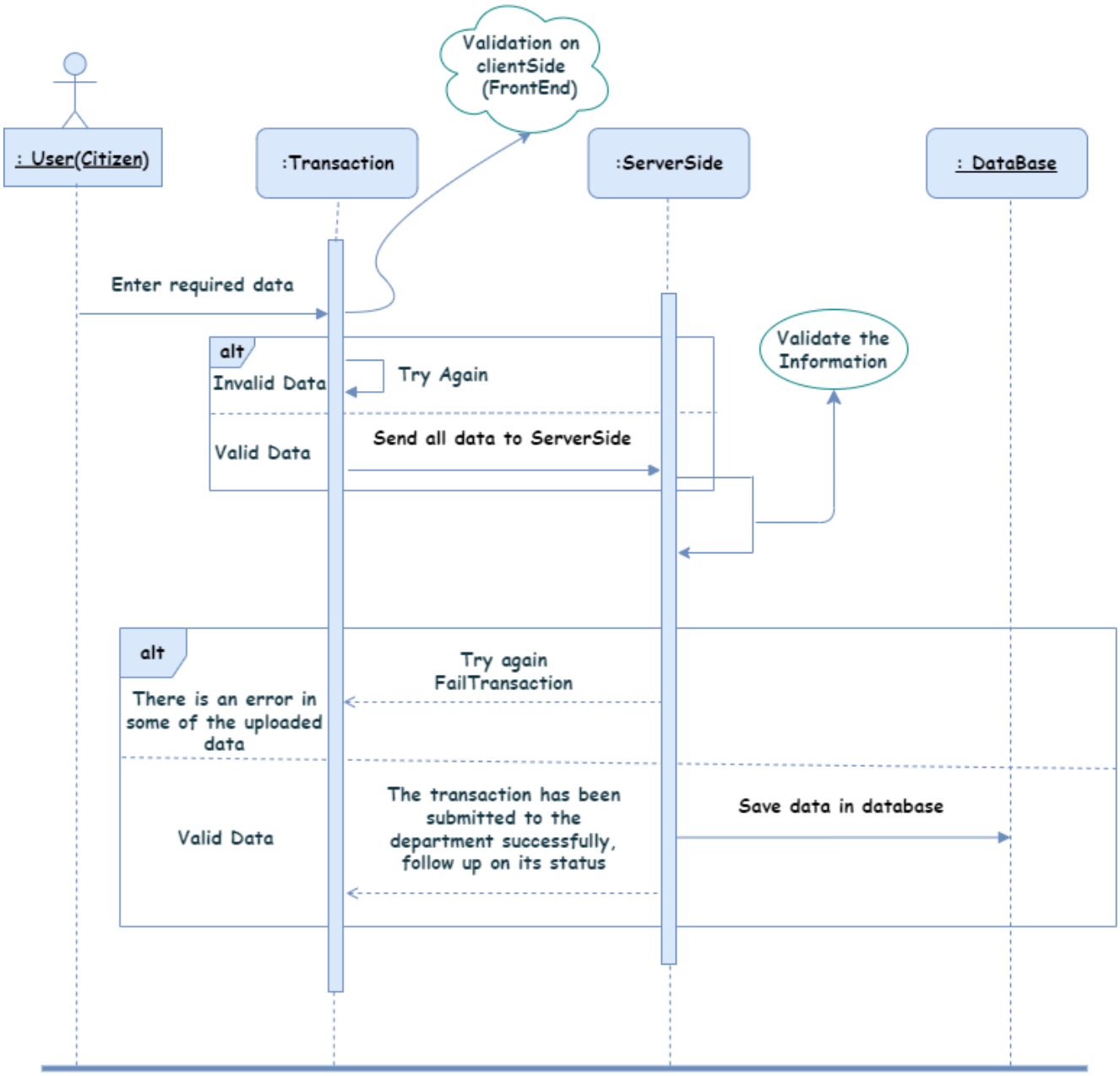


Figure 12 Transaction Sequence Diagram

4.3.3 Log-Out Sequence Diagram

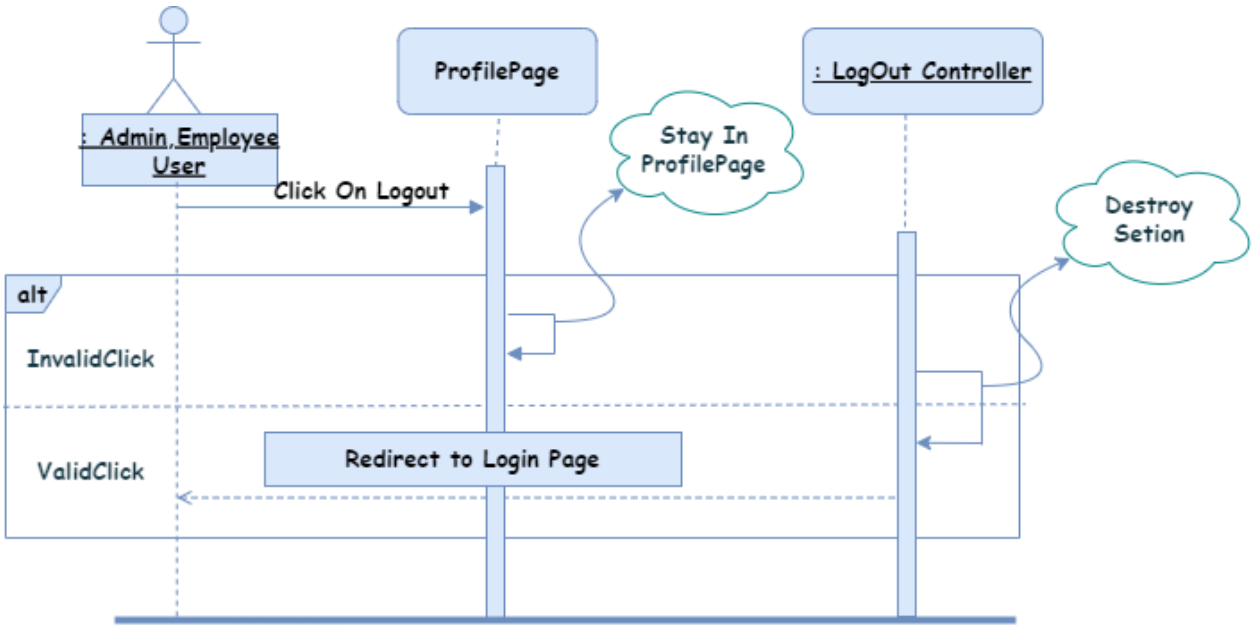


Figure 13 Log-Out Sequence Diagram

4.4 Relational Schema Diagram

A relational schema diagram, also known as an entity-relationship (ER) diagram or database schema diagram, is a visual representation of the logical structure of a relational database. It illustrates the tables/entities, their attributes/fields, and the relationships between them.

4.4.1 Entity Relationship Diagram

4.5 State Machine Diagram

A state machine, also known as a finite-state machine (FSM), is a conceptual model used in software engineering to represent the behavior of a system that can exist in different states and transition between them based on specific events or conditions.

4.5.1 Freez Employee state Machine Diagram

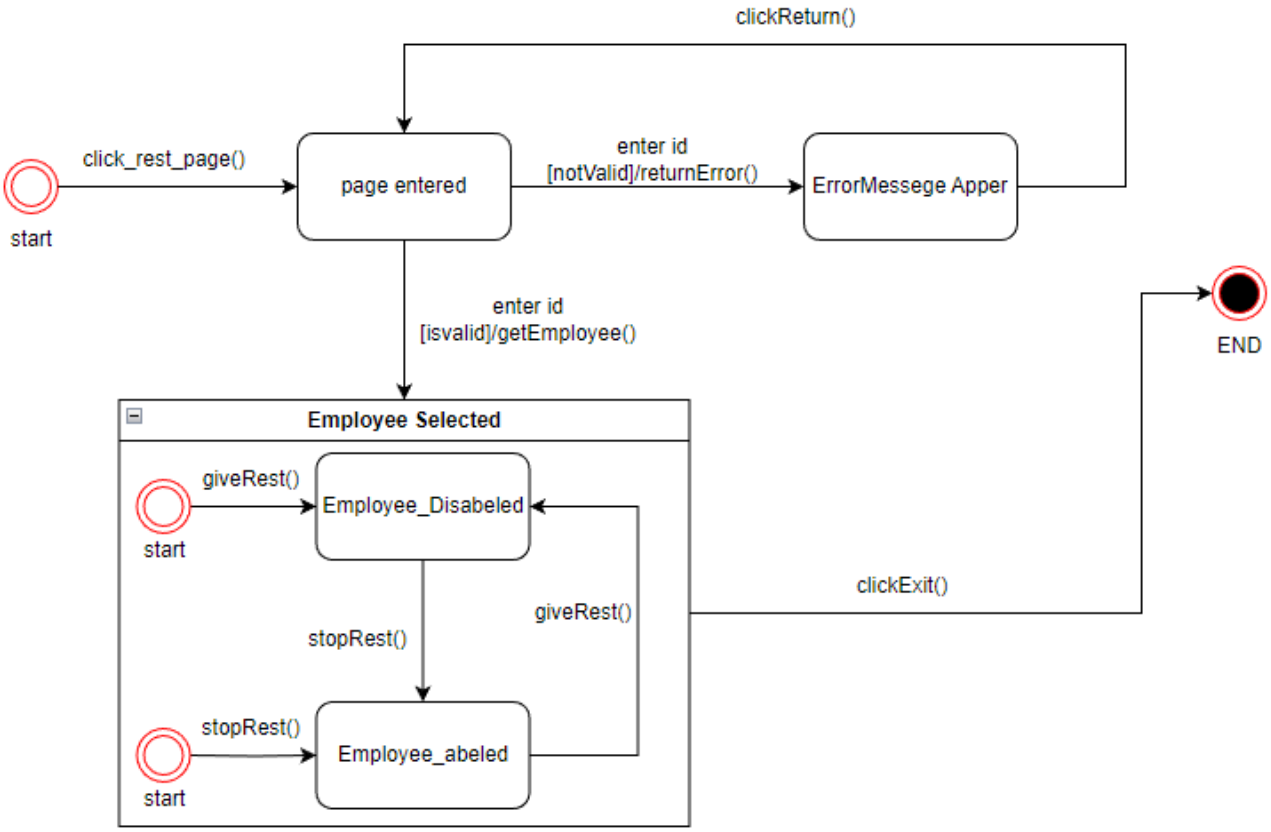


Figure 14 Freez Employee state Machine Diagram

4.5.2 Send Transaction State Machine Diagram

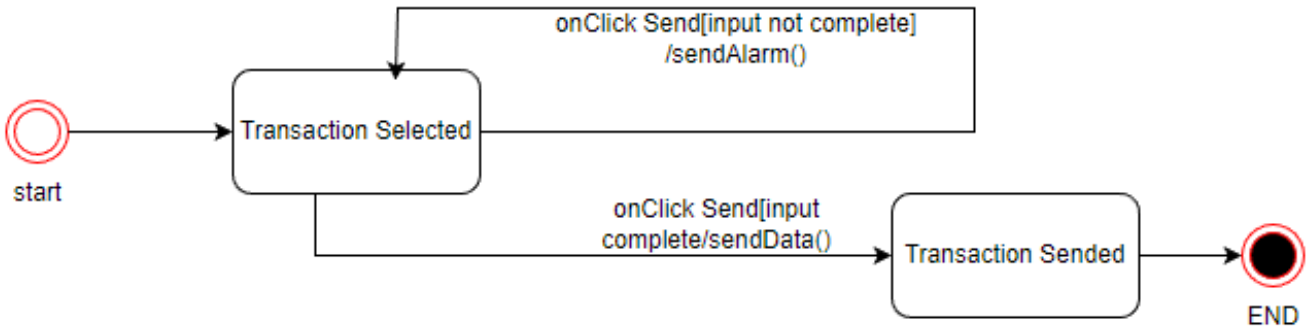


Figure 15 Send Transaction State Machine Diagram

4.5.3 Check Transaction State Machine Diagram

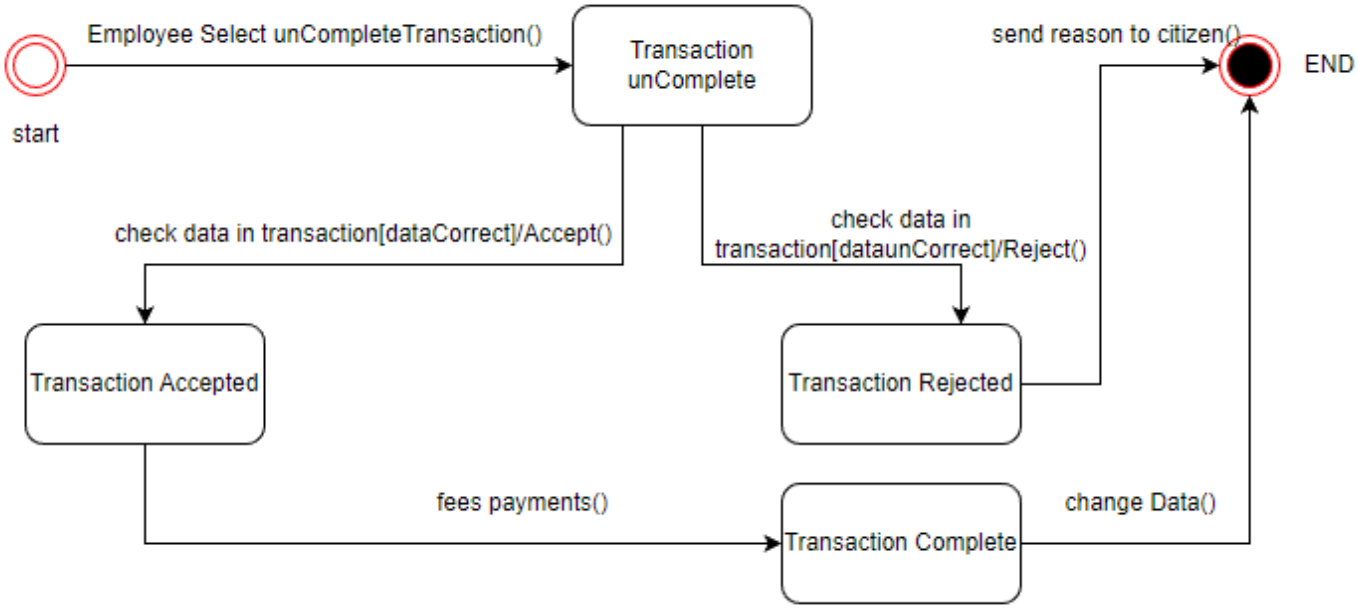


Figure 16 Check Transaction State Machine Diagram

CHAPTER 5

TECHNOLOGIES and IMPLEMENTATION

IMPLEMENTATION

4.1 Introduction

In this chapter, we will talk about the technology used to build websites from different programming languages, frameworks, and databases used….

We will use MerneStack technology to build our website

- MernStack:

MERNStack is a compilation of four different technologies that work together to develop dynamic web apps and websites

It is a contraction for four different technologies as mentioned below: [4.1]

- M – MongoDB.
- E – ExpressJS. (Back-end framework)
- R – ReactJS in Front-end (JavaScript Library for Client-Side Work)
- N – NodeJS for Back-end (Server-Side JavaScript Environment)

4.2 What is a Web Development Stack?

A tech stack is defined as the set of technologies an organization uses to build a web or mobile application. It is a combination of programming languages, frameworks, libraries, patterns, servers, UI/UX solutions, software, and tools used by its developers.

Besides, A stack can (and is meant to) be used repeatedly to develop web applications. Individual developers, as well as software companies, often specialize in a specific stack or several stacks.

As a potential client of a software company or someone interested in developing a new web or mobile app, you should definitely be interested in the web development stack of the company you wish to work with.

It refers to the technologies they specialize in and use together to develop new pieces of software. They mainly consist of NodeJS, MongoDB, ExpressJS, Angular, React, Vue, PostgreSQL, MySQL, Apache.

4.3 MVC architecture in software

4.3.1 What is the MVC?

MVC stands for Model-View-Controller, which is a software architectural pattern commonly used in designing and developing web applications. It separates an application into three interconnected components: The Model, the View, and the Controller.

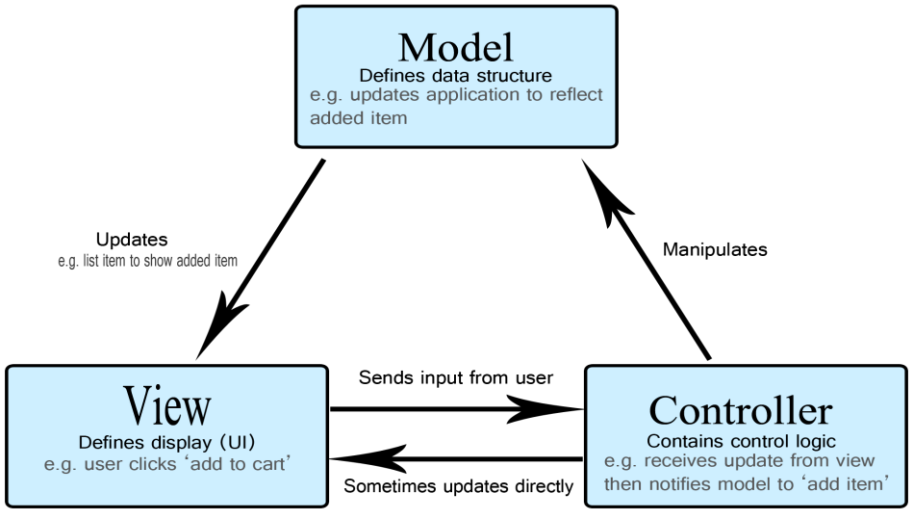


Figure 17 MVC architecture

Here's a simple abstract of each component:

1. **Model:**

The Model represents the data and the business logic of the application. It defines the structure and behavior of the data, manages data storage and retrieval, performs calculations, and enforces business rules. The Model component is independent of the user interface and does not rely on how the data is presented or accessed.

2. **View:**

The View represents the user interface of the application. It is responsible for displaying the data to the user in a visually appealing and understandable manner. The View receives data from the Model and formats it for presentation, such as rendering HTML pages or generating graphical elements. The View is passive and does not contain any business logic.

3. Controller:

The Controller acts as the intermediary between the Model and the View. It receives user input and translates it into actions to be performed on the Model or the View. The Controller handles user interactions, validates input, and updates the Model accordingly. It also selects the appropriate View to display the updated information. The Controller encapsulates the application's business logic and orchestrates the flow of data and actions between the Model and the View.

4.3.2 Why is the MVC pattern?

The key idea behind MVC is the separation of concerns, which allows for modular and reusable code. It promotes code organization, maintainability, and flexibility in software development. By isolating the data, user interface, and business logic into distinct components, changes or updates in one component can be made without affecting the others. This enables easier testing, debugging, and scalability of applications.

4.3 Front-End, Client-Side Technology Stack

A web application has two primary sides. One is available to the client and is accountable for the UI, while the other is liable for the client’s experience.

The side which is open to the client and exploited by him/her to communicate with the application establishes the front end. The front-end technology stack principally comprises HTML, CSS, Java, and so on… [4.2]

4.3.1 HTML

It is a programming language utilized for portraying the construction of data introduced on a page. Also, it utilizes the most recent rendition of HTML HTML5 which has new components and qualities for making web applications all the more effective and viably. The fundamental benefit that HTML5 is that it has sound and video support, which was excluded from past variants of HTML.

4.3.2 SASS & CSS

SASS (Syntactically Awesome Stylesheets) is a preprocessor scripting language that is interpreted or compiled into Cascading Style Sheets (CSS). It adds various features such as variables, nested rules, mixins, and functions, which can make writing CSS easier and more efficient.

CSS, on the other hand, is a stylesheet language used for describing the presentation of a document written in a markup language. It is used to apply styles, such as colors, fonts, and layouts, to web pages.

Simple example:

```
# ccss.css > div p
1   div{
2       width: 200px;
3   }
4   div p{
5       color: inherit;
6   }
```

CSS Style

```
style.scss > div
1   div{
2       width: 200px;
3       p{
4           color: inherit;
5       }
6   }
```

SASS Style

4.3.3 Bootstrap

Bootstrap is a free and open-source web development framework. It's designed to ease the web development process of responsive, mobile-first websites by providing a collection of syntax for template designs.

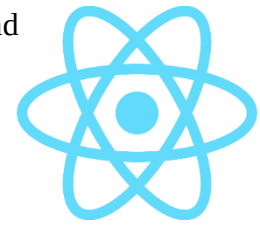
In other words, Bootstrap helps web developers build websites faster as they don't need to worry about basic commands and functions. It consists of HTML, CSS, and JS-based scripts for various web design-related functions and components. [4.3]

4.3.4 Java Script

JavaScript is a programming language that is primarily used to create interactive front-end web development. It is a high-level, dynamic, and interpreted programming language. JavaScript can be used to add interactivity and other dynamic features to websites, such as drop-down menus, form validation, and responsive design. It is also widely used for server-side programming using frameworks such as Node.js. [4.4]

4.3.5 React

The React.js framework is an open-source JavaScript framework and library developed by Facebook. It's used for building interactive user interfaces and web applications quickly and efficiently. [4.5]



- **Why React Anther Farmwork?**
- Intuitive

ReactJS is extremely intuitive to work with and provides interactivity to the layout of any UI. Plus, it enables fast and quality assured application development that in turn saves tome for both - clients and developers.

- Declarative

ReactJS enables significant data changes that result in automatic alteration in the selected parts of user interfaces. Owing to this progressive functionality, there is no additional function that you need to perform to update your user interface.

- Provides Reusable Components

ReactJS provides reusable components that developers have the authority to reuse and create a new application. Reusability is exactly like a remedy for developers. This platform gives the developers the authority to reuse the components build for some other application having the same functionality. Thereby, reducing the development effort and ensuring a flawless performance.

- JavaScript library

A strong blend of JavaScript and HTML syntax is always used, which automatically simplifies the entire process of writing code for the planned project. The JS library consists several functions including one that converts the HTML components into required functions and transforms the entire project so that it is easy to understand. [4.6]

- Market need

The frame-work is really powerful and the demand for it in the market is really high, so by learning it we are keeping up with what the market in the real world needs the most.

According to stack overflow trends, react took the lead in web developments in the last couple of years. [4.7]

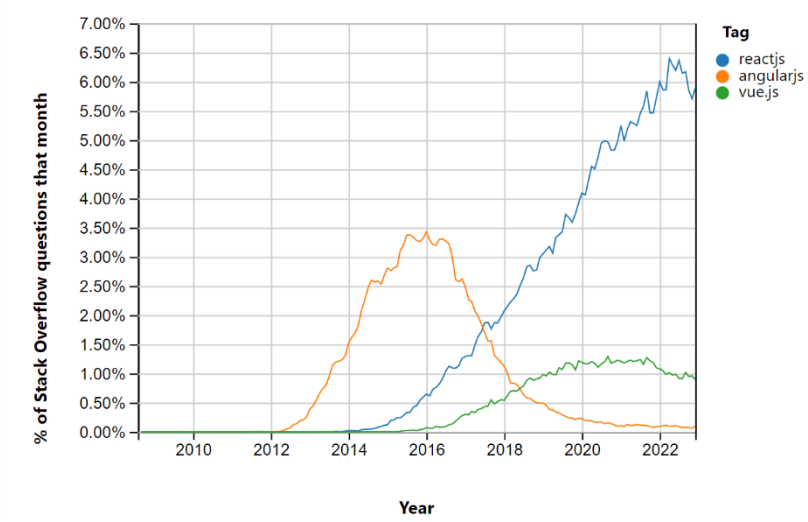


Figure 18: Comparison of frameworks in front-end

4.3.6 React-Bootstrap

React-Bootstrap is a popular open-source front-end framework for building responsive web applications using React, a JavaScript library for building user interfaces. It combines the power of React with the styling capabilities of Bootstrap, a widely-used CSS framework.

React-Bootstrap provides a set of pre-designed UI components that are ready to use in your React applications. These components include buttons, forms, navigation bars, modals, tooltips, grids, and many more. They are built with Bootstrap's CSS classes and styles, making it easy to create visually appealing and consistent UIs.

The main advantage of using React-Bootstrap is that it integrates seamlessly with React's component-based architecture. You can easily import and use React-Bootstrap components in your React code, and they can be customized and extended using React's props and state.

4.4 Back-End, Server-Side Technology Stack

The working of the application dependent on the client association is primarily the back end or server-side. It isn't available to the client, and the technology answerable for creating it is known as the back-end technology stack. Programming languages, libraries, structures, servers, data set administration frameworks, and so forth, are the huge parts of back-end tech stacks. [4.8]

4.4.1 NodeJs

Node.js is an open-source, cross-platform JavaScript runtime environment that executes JavaScript code outside of a web browser. It allows developers to use JavaScript on the server side to build fast and scalable network applications. Node.js uses an event-driven,

non-blocking, I/O model, which makes it lightweight and efficient. It is commonly used for real-time applications, such as chat and gaming applications, as well as for building web servers and APIs.

4.4.2 ExpressJs

Express JS is a Node.js framework designed to build API's web applications cross-platform mobile apps quickly and make node js easy.

- **What Is Express JS?**

Express is a node web application framework that provides broad features for building web and mobile applications. It is used to build a single page, multipage, and hybrid web application.

It's a layer built on the top of the NodeJS that helps manage servers and routes.

- **Why Express JS?**

i. Express was created to make APIs and web applications with ease, it saves a lot of coding time almost by half and still makes web and mobile applications are efficient.

ii. Another reason for using express is that it is written in JavaScript as JavaScript is an easy language even if you don't have a previous knowledge of any language. Express lets so many new developers enter the field of web development.

iii. The reason behind creating an express framework for NodeJS is:

1 Time-efficient
2 Fast
3 Economical
4 Easy to learn
5 Asynchronous. [4.9]

Simple server using express.js

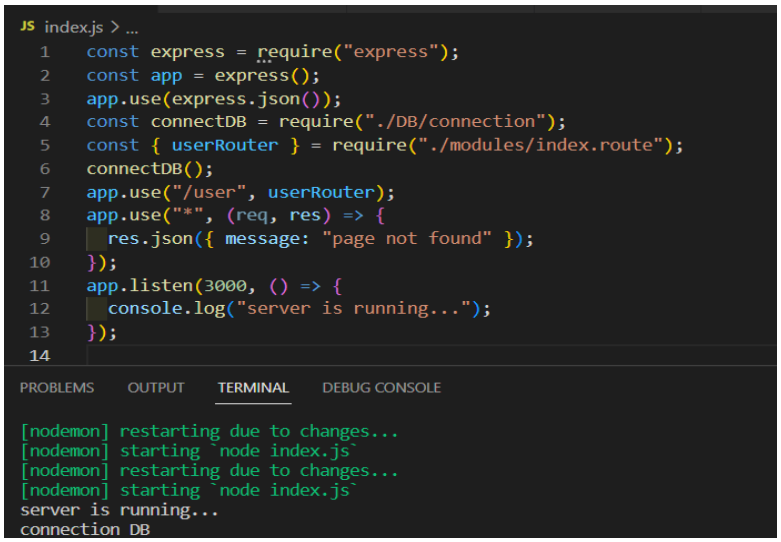


Figure 19: Example using Express.Js

4.5 Database

A database is an organized collection of structured information, or data, typically stored electronically in a computer system. A database is usually controlled by a database management system (DBMS). Together, the data and the DBMS, along with the applications that are associated with them, are referred to as a database system, often shortened to just database.

4.5.1 What is a NoSQL database?

When people use the term “NoSQL database,” they typically use it to refer to any non-relational database.

Some say the term “NoSQL” stands for “non SQL” while others say it stands for “not only SQL.” Either way, most agree that NoSQL databases are databases that store data in a format other than relational tables [4.10]

4.6 Web Development Environment

4.6.1 Visual Studio Code (VSCode)

To implement our TABO project, we use Microsoft Visual Studio code 2022 to develop the site.

Visual Studio Code (often shortened to VS Code) is a free, open-source code editor developed by Microsoft for Windows, Linux, and macOS. It is built on the Electron framework and uses the same core components as the full Visual Studio IDE, but with a more lightweight and customizable user interface.

VS Code is designed to be a lightweight, fast, and efficient code editor that can handle a wide variety of programming languages and file formats. It supports built-in debugging, integrated Git control, syntax highlighting, and IntelliSense (smart code completion), and offers a wide range of extensions that can add additional functionality, such as language support, themes, and debugging tools.

VS Code is widely used by developers for web development, front-end development, mobile development, and other types of programming, it's also a popular tool for data scientists and analysts, it is widely accepted for its speed, flexibility, and variety of supported languages and platforms. [4.11]

Recap

In this chapter, we talked about the modern techniques and technology that we will use to build our project, and we clarified some concepts and differences between the different frameworks and why we chose each of them.

CHAPTER 6

RESULTS

6. System Interfaces

In this section, we will present images showcasing the outcomes achieved upon the completion of the system's development, indicating its readiness for deployment.

6.1 Admin Interfaces

The main page that will appear for the admin (Manager of registry land) when logging in to interact with the system.

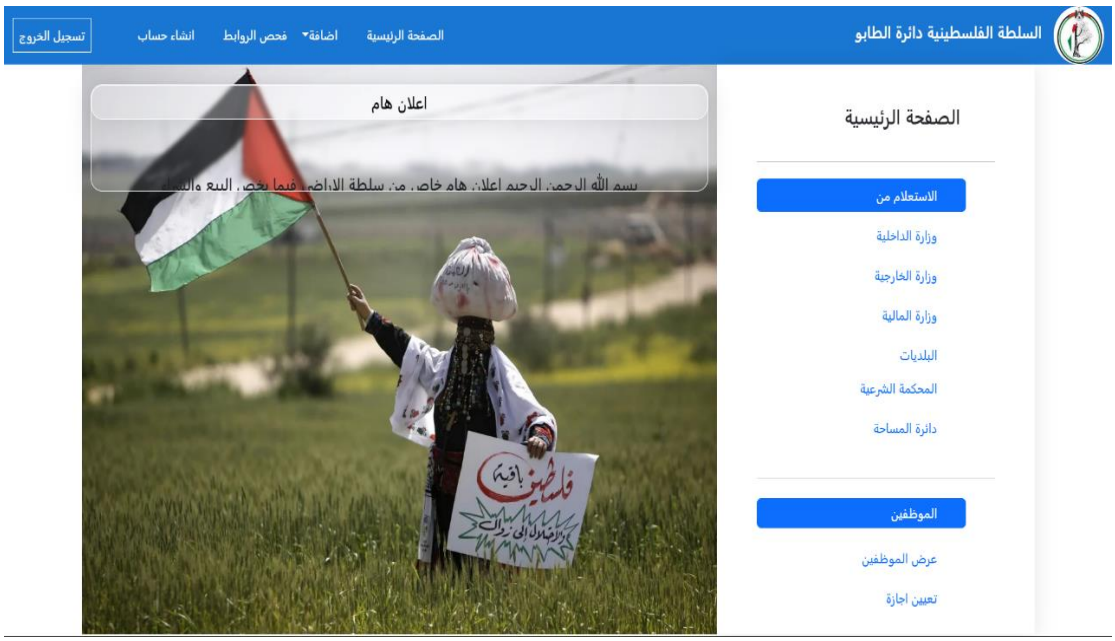


Figure 20 Admin Dashboard

The page that will appear to the admin when he wants to create an account for a new employee within the registry land.

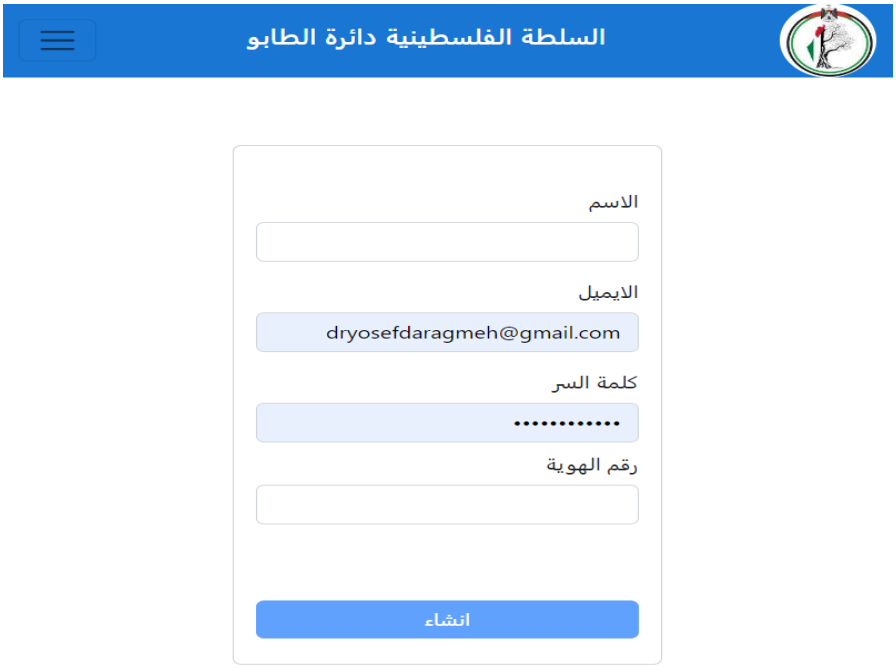


Figure 21 Create Account for Employee

This page displays the department's employees along with the tasks assigned and completed by each individual, enabling the administrator to effectively monitor the employees.

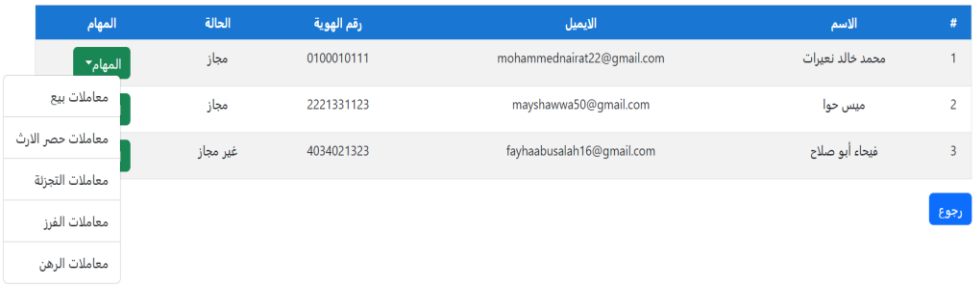


Figure 22 Employee Workflow

This page shows the sales transactions assigned to a specific employee, indicating their responsibility in handling those transactions.

Figure 23 Sale Transaction for an Employee

This page shows that the admin can inquire about an citizen in the Ministry of Interior via the ID number.

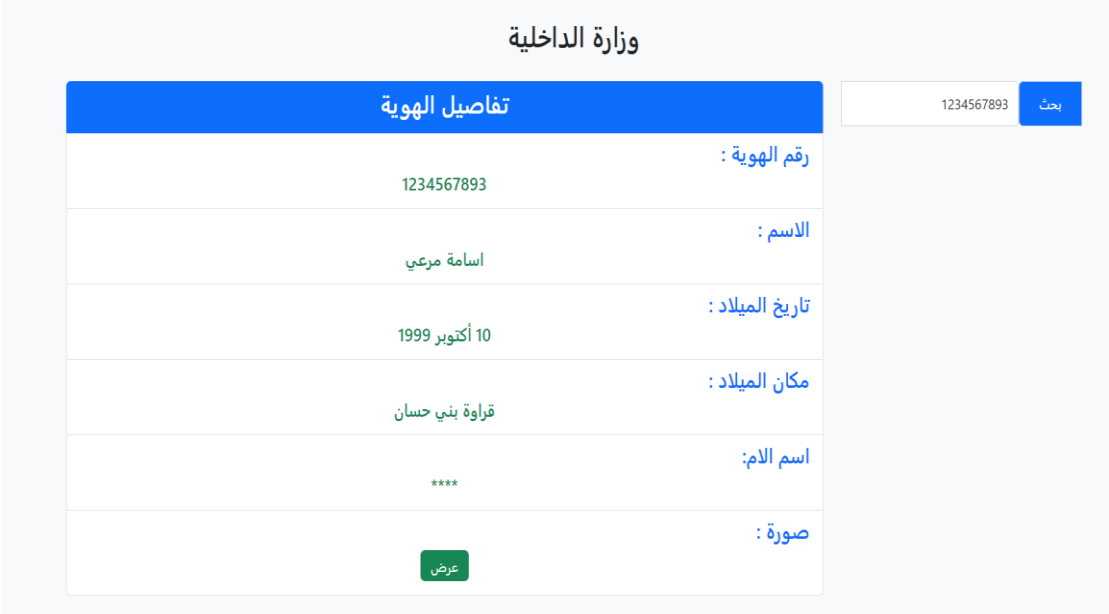


Figure 24 Enquire from Ministries

This page shows that the admin can add and publish news that appears dedicated to employees so that it appears on their pages, or for citizens in the same case.

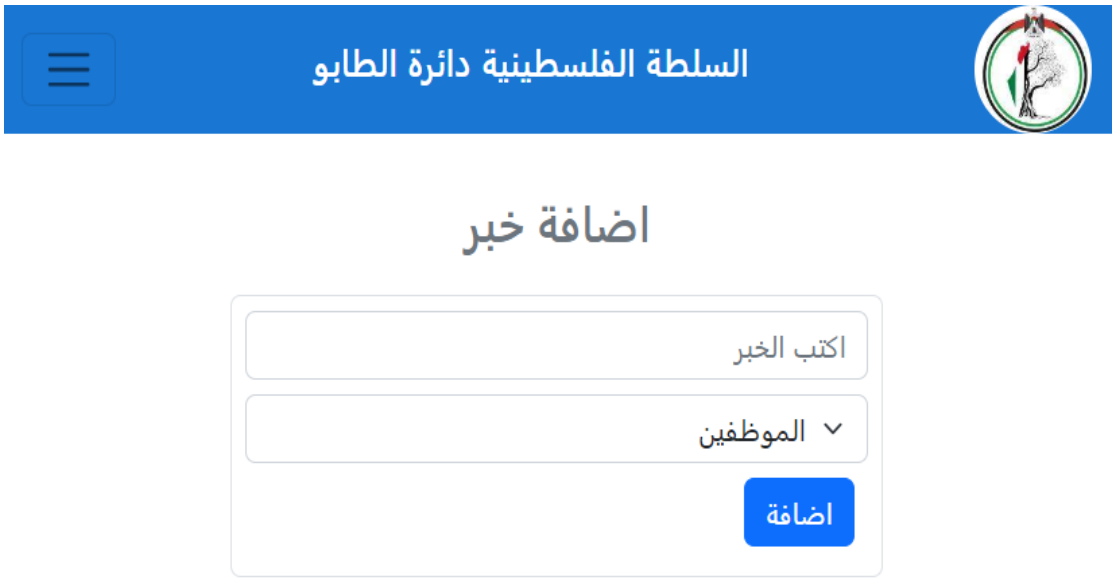


Figure 25 Add News for Employee or Citizen

Admin can check links; Whether the link is valid or fake. **[7.1]**

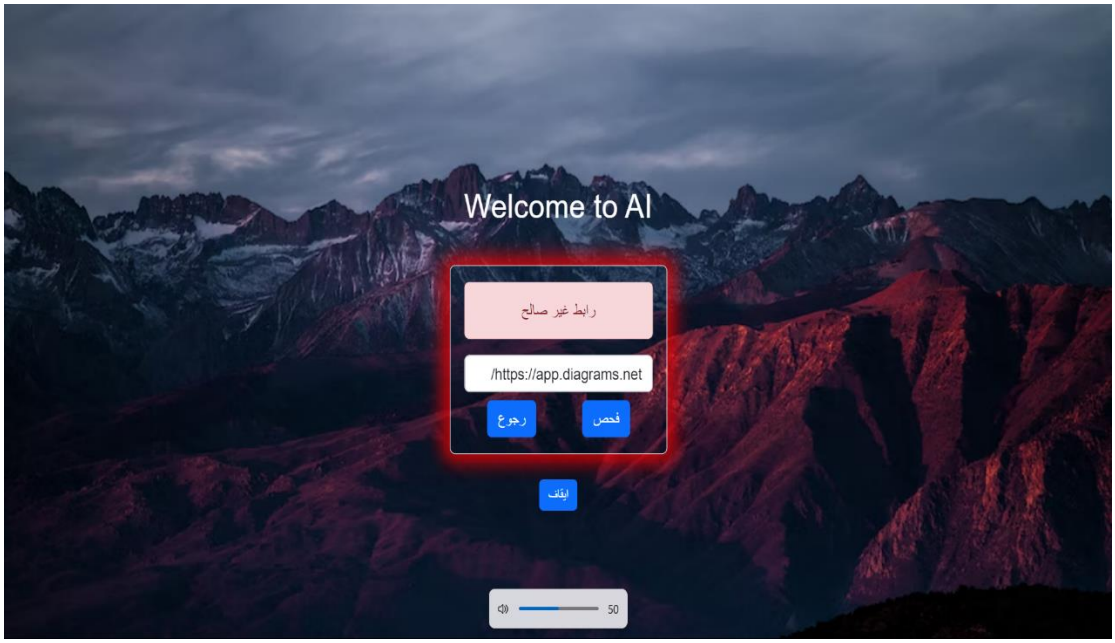


Figure 26 Check Links

This page shows that the admin can inquire about an employee and give vacation or cancel it.

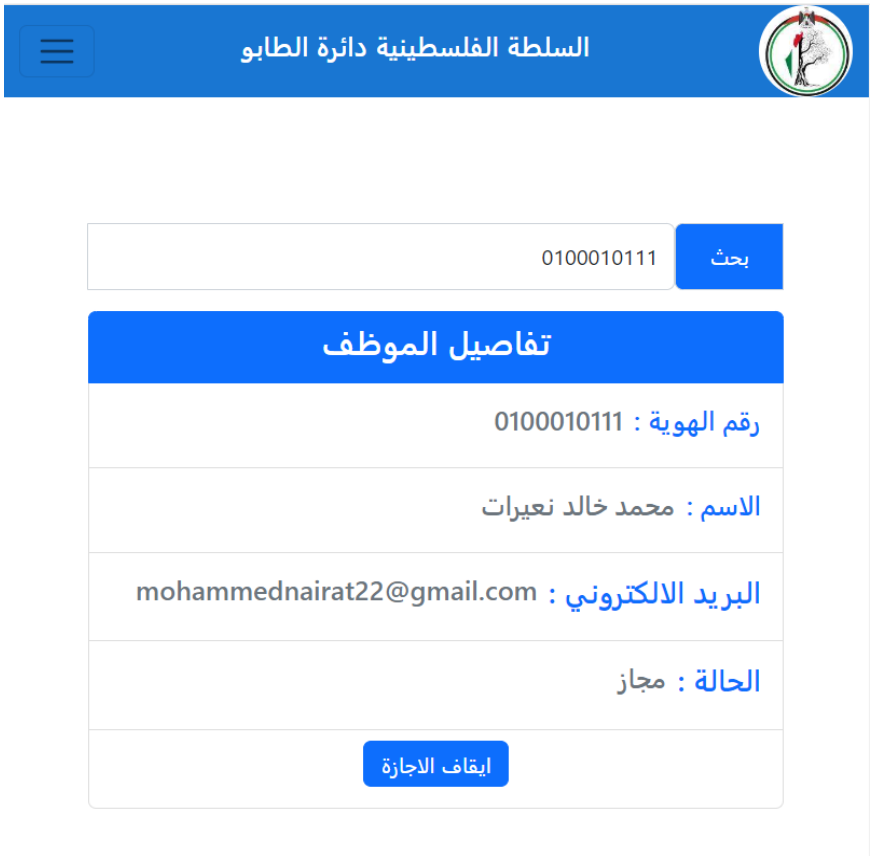


Figure 27 Give Vaccation

This page shows that the admin can inquire about land for an citizen in the area department.

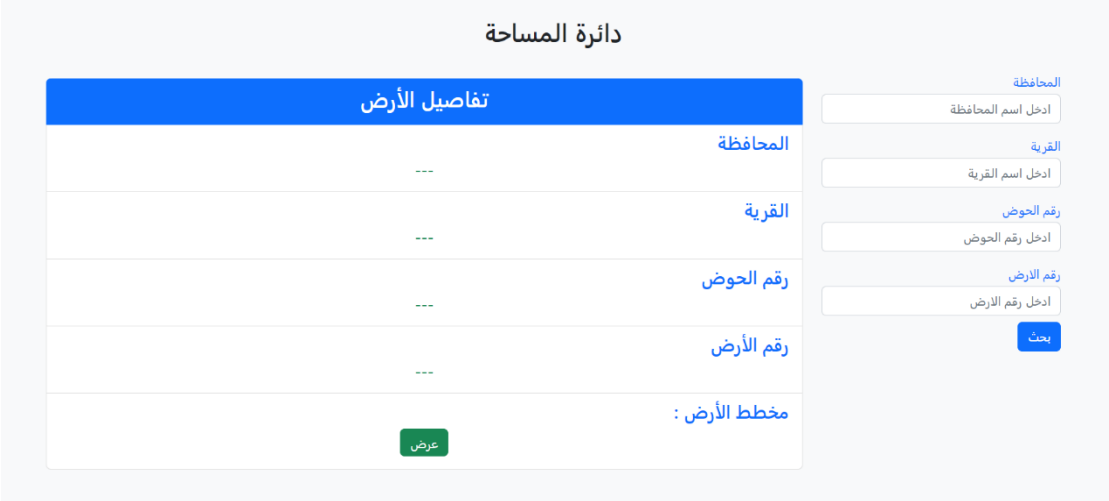


Figure 28 Inquire from Area Department

a. Employee Interfaces

The main page that will appear for the employee (Employee in registry land) when logging in to interact with the system

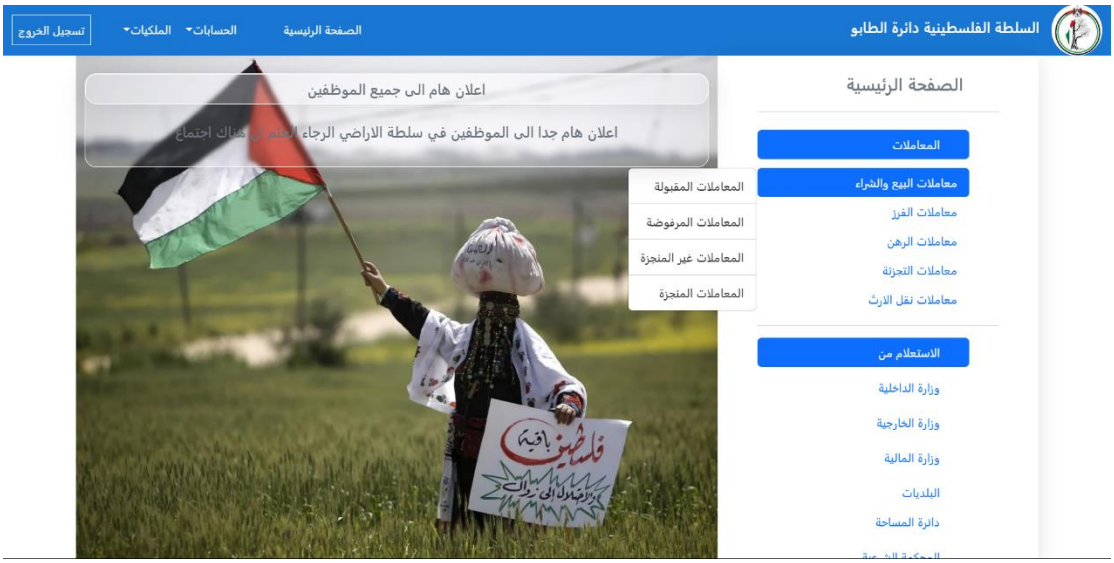


Figure 29 Employee Dashboard

This page appears to the employee when he needs to amend a citizen's data by entering the citizen's ID number

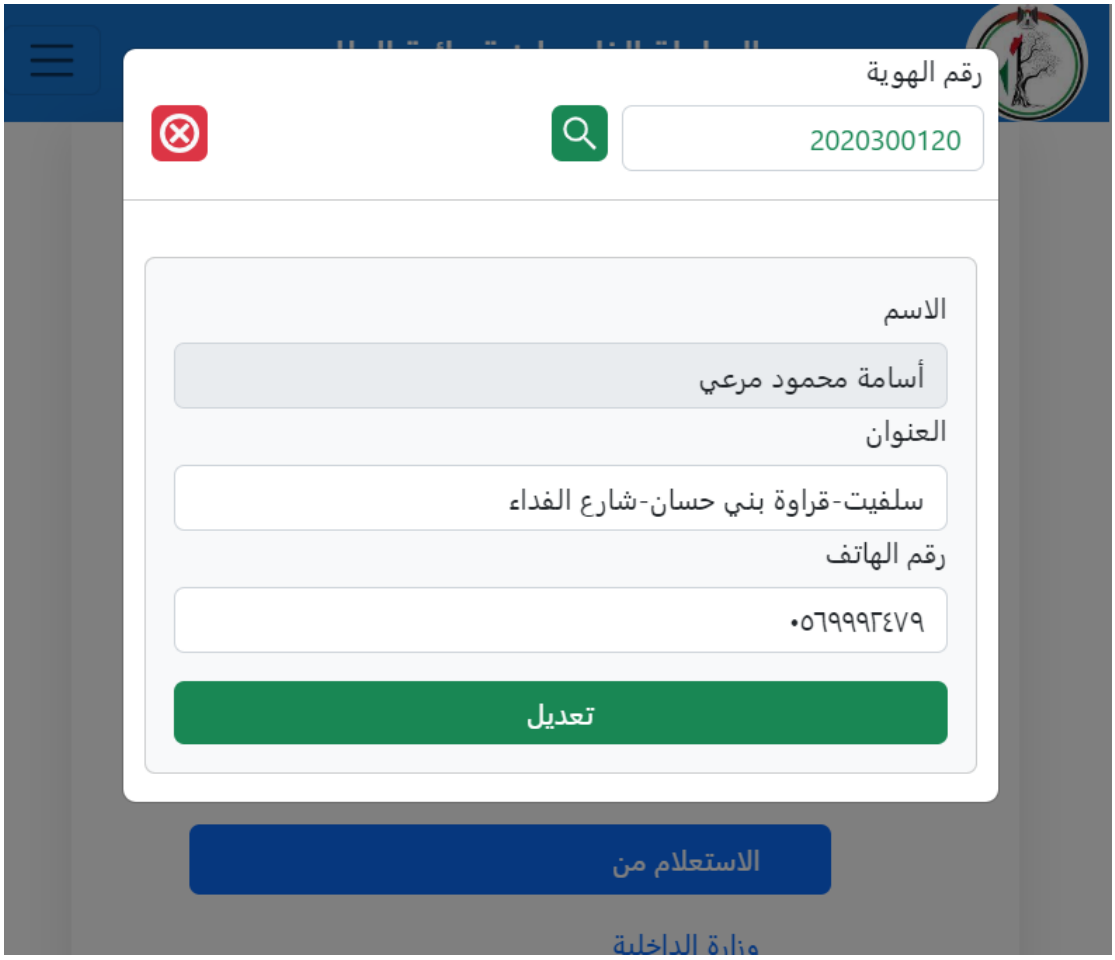


Figure 30 Edit User Data

This page appears to the employee when viewing uncompleted sales transactions, and there are 20 types of this page

Figure 31 Citizen Transaction

This page shows the details of the uncompleted transaction and the employee can accept it or reject it

Figure 32 Incompleted Transaction

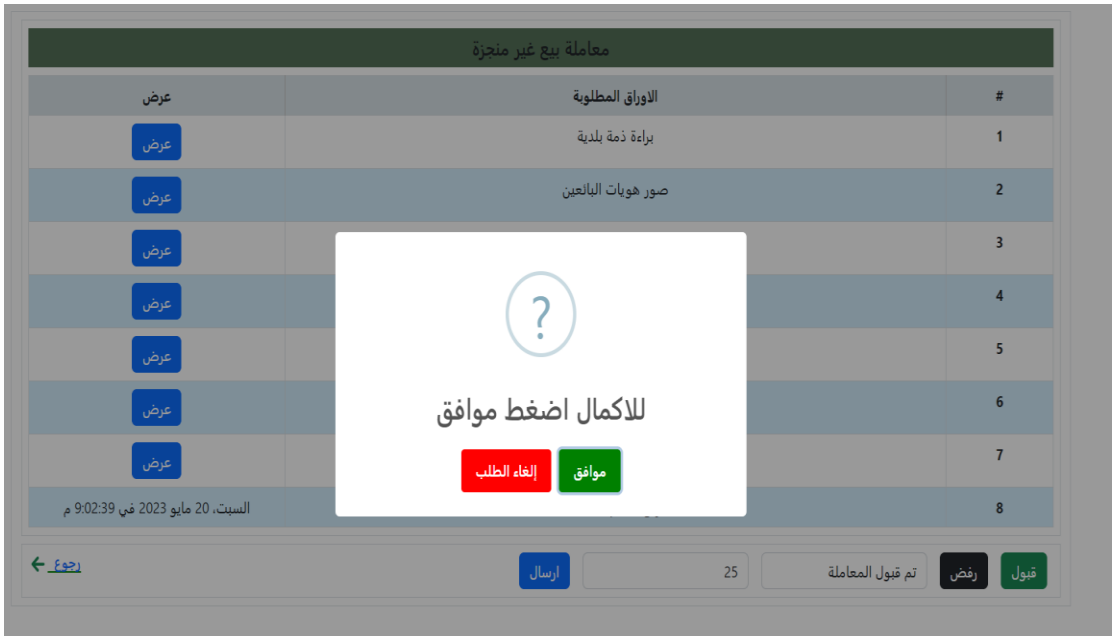


Figure 33 Accepted Tranaction and Click Send

Figure 34 Accepted Tranaction

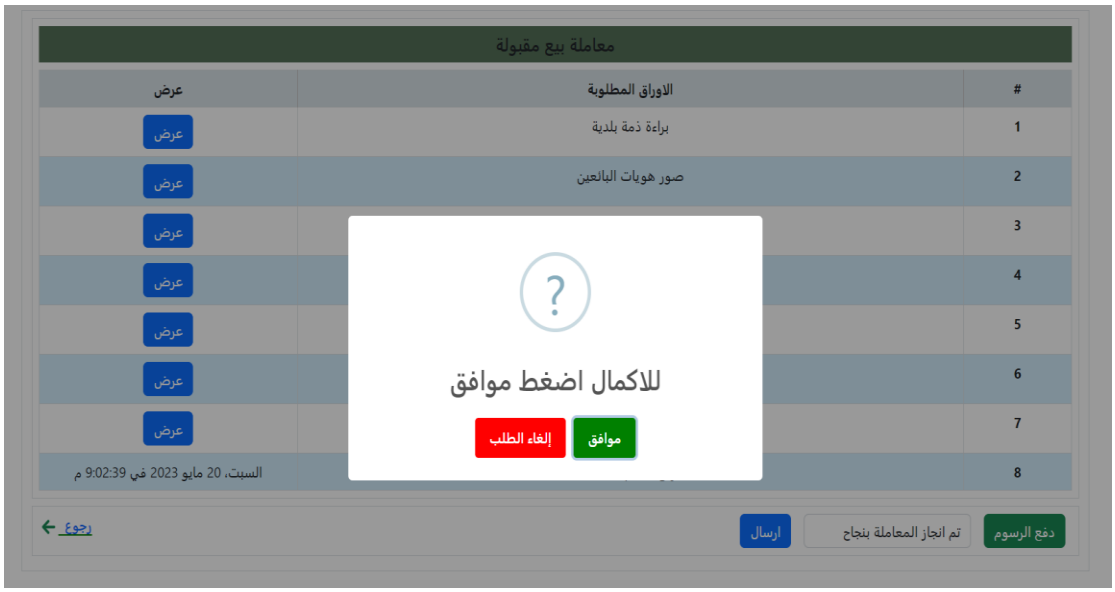


Figure 35 Paid Fees

b. Citizen Interfaces

The main page that will appear for the citizen when logging in to interact with the system

Figure 36 Citizen Dashboard

This page allows the citizen to complete a transaction and upload it to the system

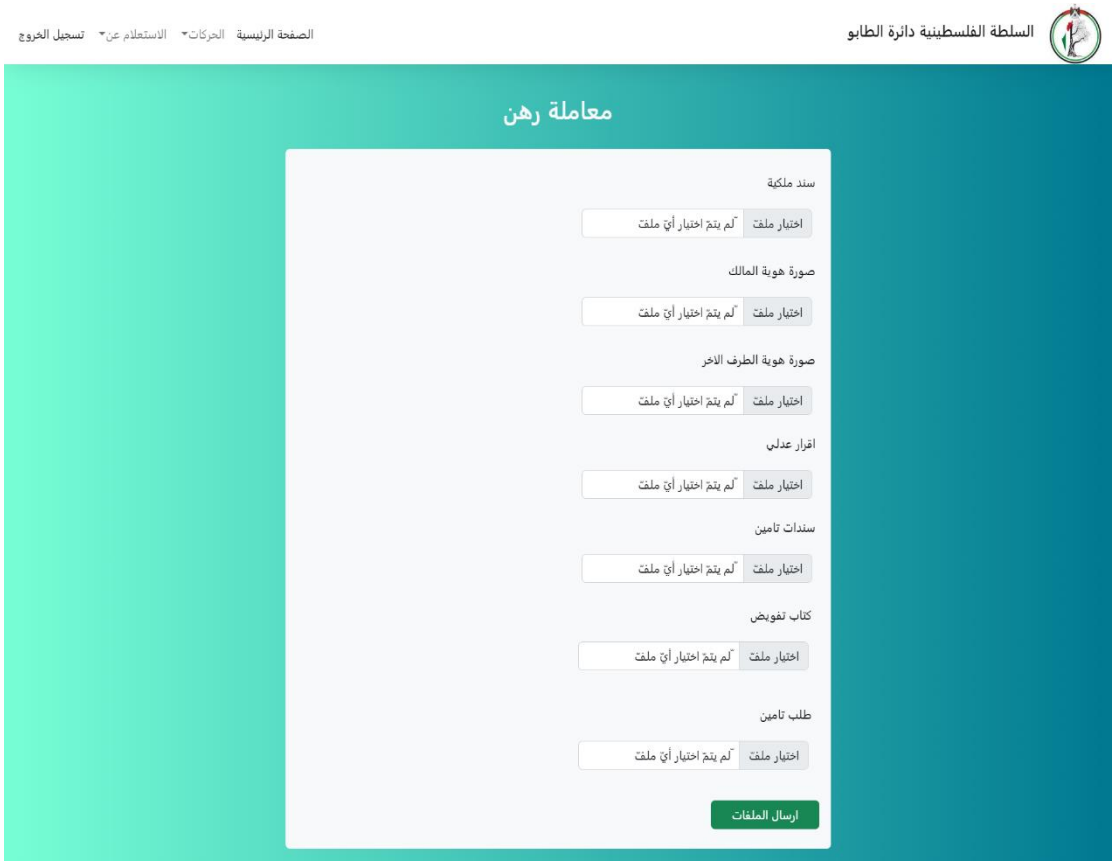


Figure 37 Mortgage Tranaction

This page appears to the citizen when the transaction is completed successfully

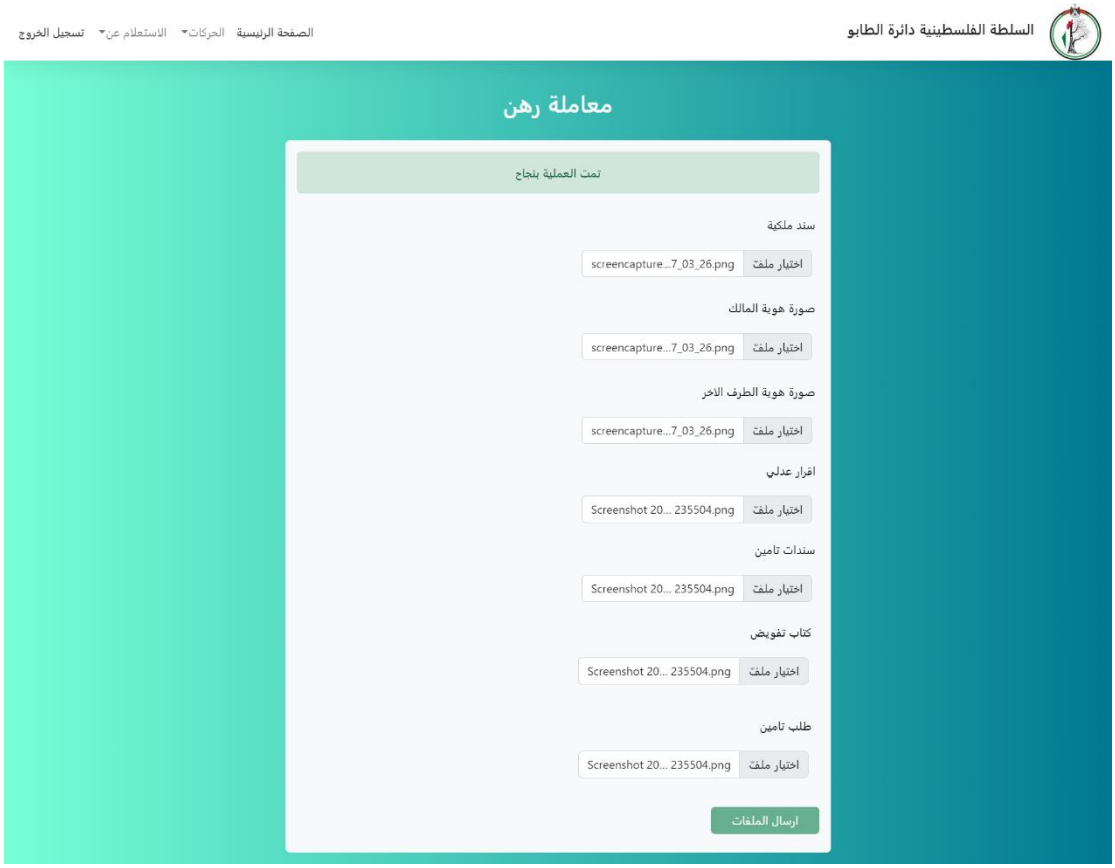


Figure 38 Submit Tranaction

This page appears to the citizen when inquiring about his land and viewing his land in the form of a PDF file

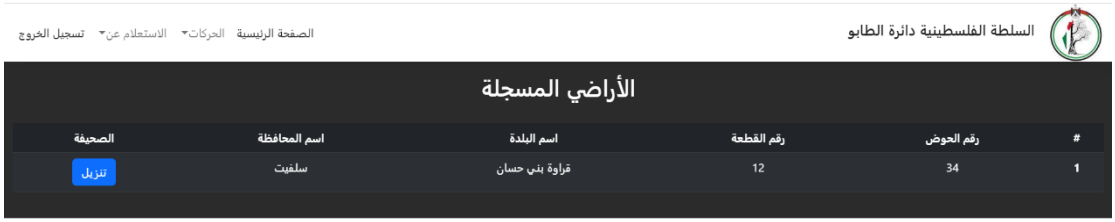


Figure 39 Enquire about Land

This page appears to the citizen when inquiring about his transactions

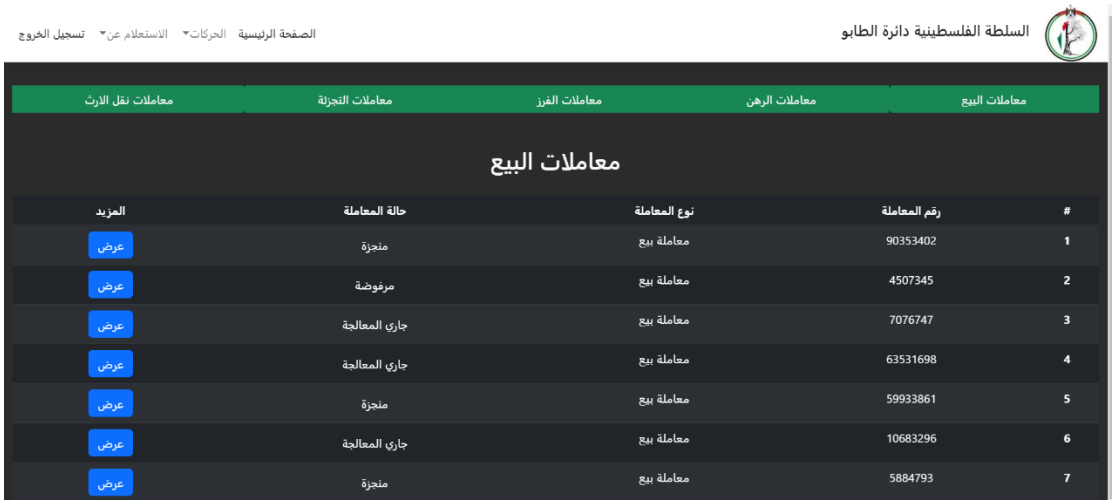


Figure 40 Enquire about Tranaction

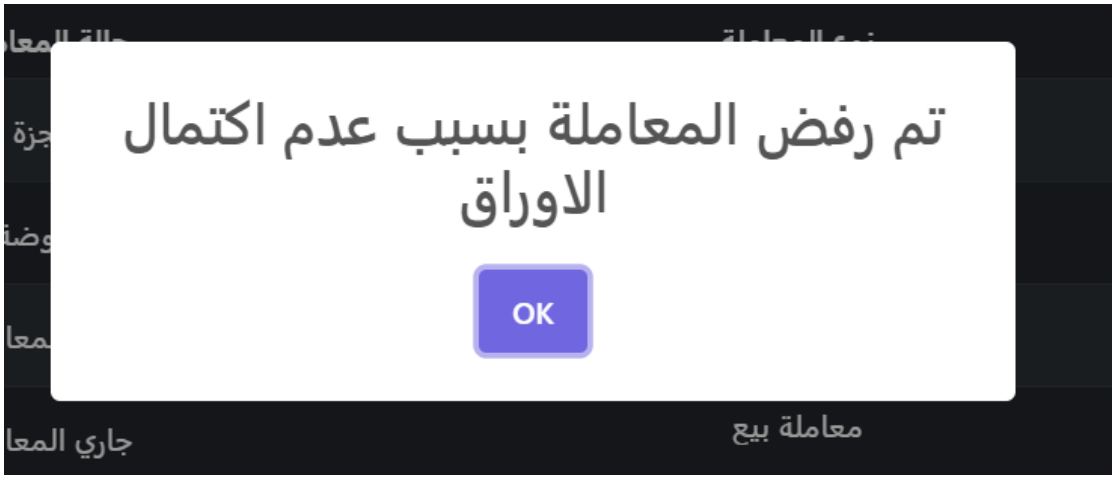


Figure 41 Show Message For a Tranaction

Figure 42 Show Transaction and Edit It

6.4 Authentication

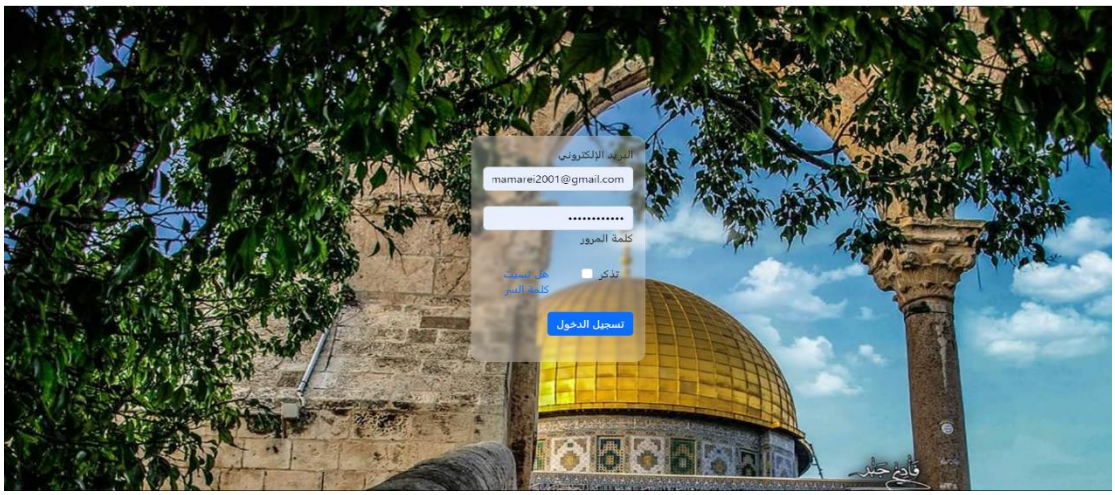


Figure 43 Login Page

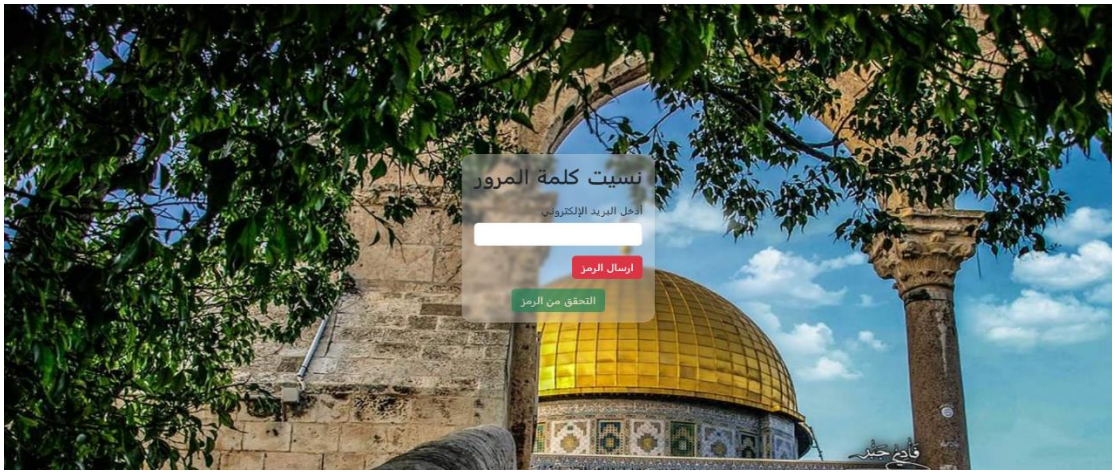


Figure 44 Enter Email When Forget Password

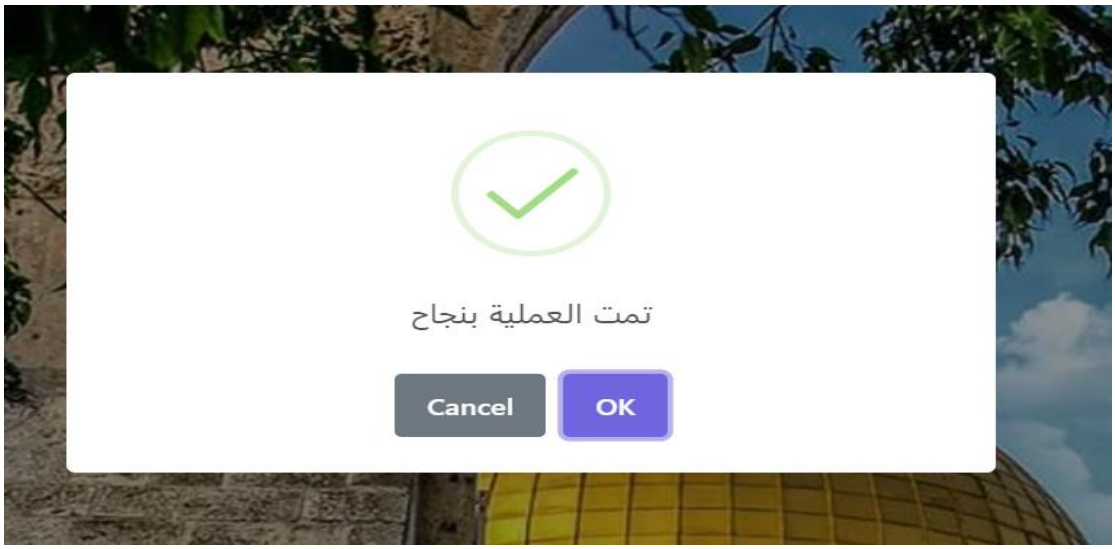


Figure 45 Success Send Message

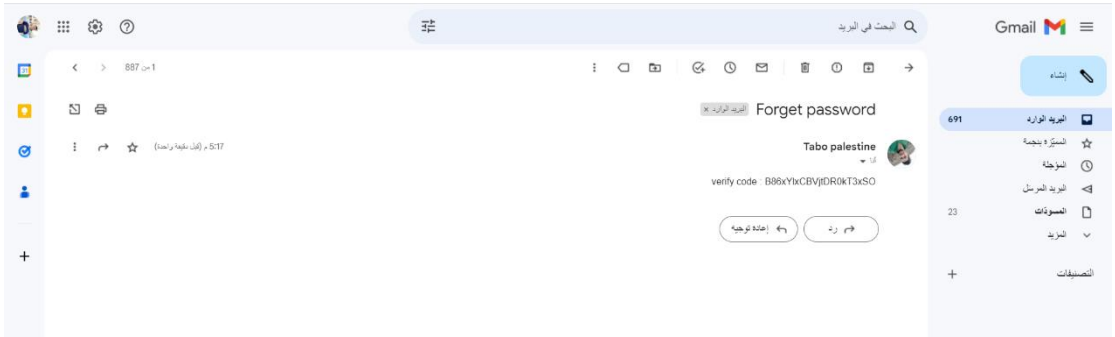


Figure 46 The Code Has Arrived to My Email

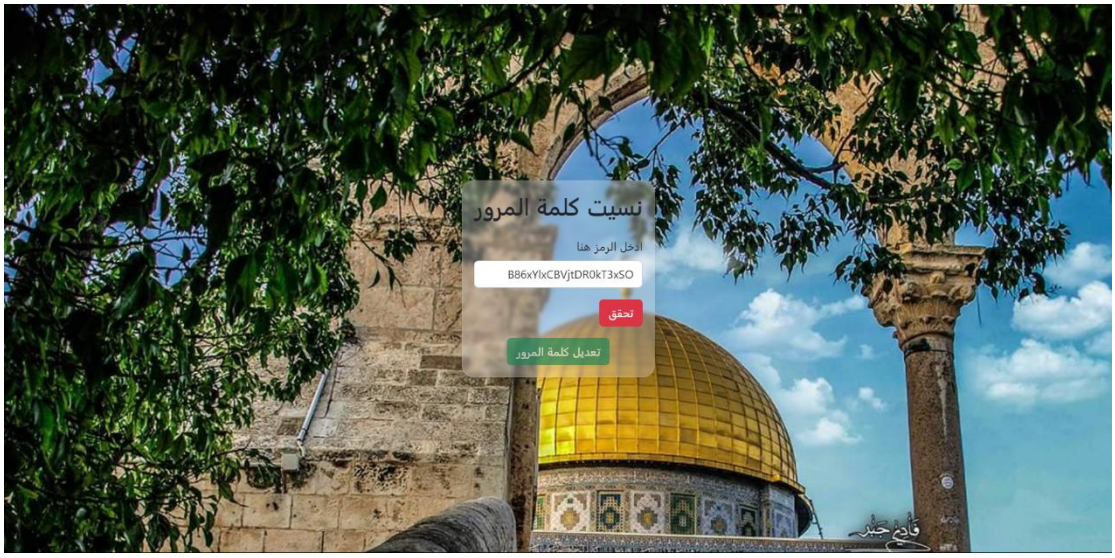


Figure 47 Inert Code to Verifying

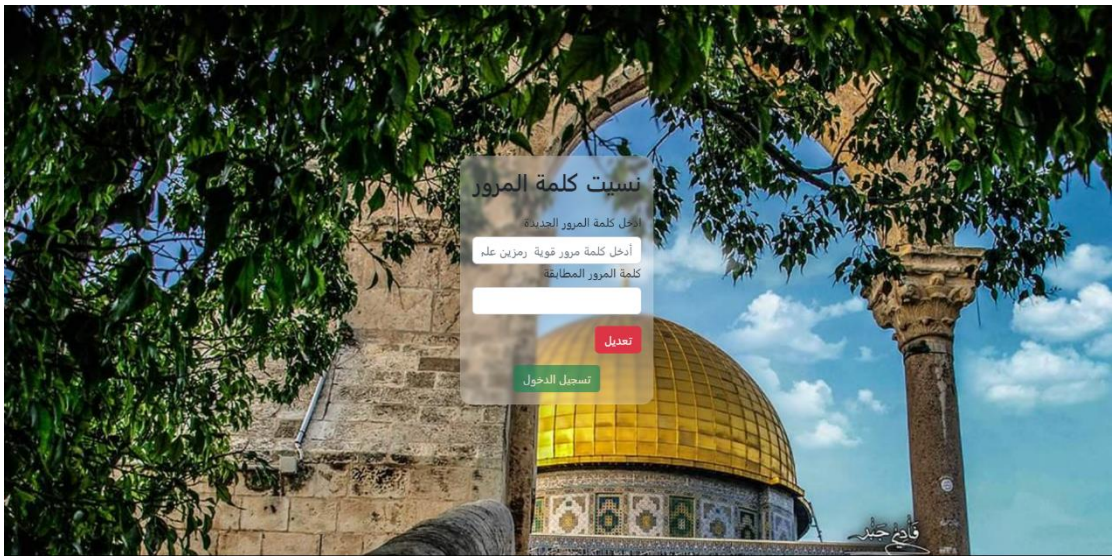


Figure 48 Insert New Password

CHAPTER 7

CONCLUSION and FUTURE WORK

7.1 Conclusion

We the youth of Palestine, who always strive to serve our land, present this project in the hope of developing and adopting it to help our country develop. At the present time, the Palestinian citizen suffers from the difficulty of completing private transactions on his land, such as selling, mortgaging, etc.

Because this requires him to visit government institutions to bring all the papers related to his transactions and review them with the land registry. Some errors may occur in these papers or they may be damaged, so he has to do them again, except for the time wasted in coming to the land registry and waiting for a period of time to obtain a service.

We have developed the initial version of the system so that it allows the citizen to submit his transactions and complete them electronically without the need to access the Land Authority except once during the payment process, so that this also makes it easier for employees to make the process smoother and keep data in a smooth manner.

In summary, the Palestinian Tabou Website is a pioneering experience in the Palestinian Society because of its positive effects on the citizens for their convenience, and it will also lead to an increase in the efficiency of work in the Tabou Department. In general, the system is a good step forward in order to achieve the development of our country, Palestine.

7.2 Future Vision

1. Adding New Features

Linking through this website the Land Authority with other governmental institutions that the citizen needs to implement his land procedures, such as the Ministry of Interior, the Ministry of Foreign Affairs, and others. Thus, the citizen will not need to visit these government institutions to bring papers related to a land transaction because the employee will be able to access them through the site and make sure of all procedures, and making the payment process electronically computerized, which allows the citizen to complete his transactions in full without accessing the land registry, as this will reduce the effort for the citizen And it will make it easier for the employee to work on documents and data stored electronically, so access to them will be faster.

2. Deploying The System

We hope that one day we deploying the system and start providing the service that established for, once that happens, we will take the responsibility of providing the necessary support to make everything get smoothly works for all users, and we are ready to add any features that users want to be available in the system, which would give the user a better experience when using the system.

CHAPTER 8
REFERENCES

[2.1] Jordanian Department of Land and Surveys Website (DLS)

https://www.dls.gov.jo/ar/Pages/default.aspx [Accessed 11-June-2023]

[2.2] The Digital Portal for Government Services
https://ssoidp.gov.ps/sso/module.php/core/loginuserpass.php?AuthState=_0d3e2d9df21ceec50a8f7eea7a0056cca190811a4f%3Ahttps%3A%2F%2Fssoidp.gov.ps%2Fsso%2Fsaml2%2Fidp%2FSSOService.php%3Fspentityid%3Dhttps%253A%252F%252Fe.services.gov.ps%252Fsso%252Fwww%252Fmodule.php%252Fsaml%252Fsp%252Fmetadata.php%252Fdefault-sp%26cookieTime%3D1686445977%26RelayState%3Dhttps%253A%252F%252Fe.services.gov.ps%252Fapps%252Favailable%253Fcitizens_ser%253D1%2526serviceName_ser%253D%2526ministry_ser%253D-1 [Accessed 11-June-2023]

[4.1] All You Need to Know About MERN Stack
https://www.simplilearn.com/tutorials/mongodb-tutorial/what-is-mern-stack-introduction-and-examples [Accessed 16-Jan-2023]

[4.2] Web Development Stack

https://dev.to/theme_selection/best-web-development-stack-2jpe [Accessed 16-Jan-2023]

[4.3] What is Bootstrap?

https://www.hostinger.com/tutorials/what-is-bootstrap/ [Accessed 16-Jan-2023]

[4.4] Comparison between Java & JavaScript

https://www.spiceworks.com/tech/devops/articles/java-vs-javascript/ [Accessed 16-Jan-2023]

[4.5] React (JavaScript library)

https://reactjs.org/ [Accessed 13-Jan-2023]

[4.6] Advantages of ReactJS for Building Interactive User Interfaces

https://www.clariontech.com/blog/7-advantages-of-reactjs-for-building-interactive-user-interfaces [Accessed 13-Jan-2023]

[4.7] Stack Over Flow

https://insights.stackoverflow.com/trends?tags=reactjs%2Cangularjs%2Cvue.js

[4.8] Back-End, Server-Side Technology Stack

https://dev.to/theme_selection/best-web-development-stack-2jpe

[4.9] What Is Express JS In Node JS?

https://www.simplilearn.com/tutorials/nodejs-tutorial/what-is-express-js [Accessed 13-Jan-2023]

[4.10] What is a NoSQL Database

https://www.mongodb.com/nosql-explained

[4.11] Visual Studio Code

https://code.visualstudio.com/docs [Accessed 14-Jan-2023]

[7.1] [Applied computing department graduation project-team leader: Ahmad Saleh]

[https://github.com/AhmadSaleh2001/PhishingURLDetection.git]

Books

1-Pro MERN Stack, Second Edition, Auther: Vasan Subramanian

2- Full-Stack React Projects, Second Edition, Auther: Shama Hoque