Software Requirements Specifications

UNDER THE SUPERVISION OF
Dr. Tarek Elghazaly

Name	ID
Salma Ayman	19104941
Alaa Mamdouh	19106209
Khalil Ahmed	19100923
Ziad Tamer	19104259
Omar Alshafaey	19100227

Table of contents

Table of contents 2

Table of figures 3

1. Introduction 4

1.1. Purpose.....4

1.2. Scope 4

1.3. Technologies used.....4

1.4. ntended audience.....4

1.5. References.....5

1.6. Overview.....5

2. Overall Description 5

2.1. Product Perspective.....5

2.2. User Characteristics 6

2.3. Operating Environment.....6

2.3. Constraints 7

2.4.Assumptions and Dependencies 7

3. Interfaces 7

3.1. System Interface.....7

3.2. Software Interface.....10

3.3. Hardware Interface.....10

4. Functional Requirements.....11

4.1 User Class 1 - The User 11

4.2. User Class 2 - The Admin 15

5. Non-Functional requirements 15

6. Diagrams 18

Table of figures

Figure 1 loading Screen 8

Figure 2 log in page 8

Figure3 Available services 8

Figure4 Service sub-menu 8

Figure5 9

Figure6 9

Figure 7 9

Figure 8 9

Figure 9 10

Figure 10 10

1. Introduction

1.1. Purpose

The aim of this document is to provide a comprehensive overview of People's app system. It will describe the stem's function and features, as well as the system's interfaces, what the system will do, the constraints under which it must work, and how the system will respond to external stimuli

1.2. Scope

People’s app is an app that provides easy way to make the government papers by installing everything about that citizen from his/her date of birth until date of death on a large database. The government are going to have this database to check the citizens papers without even ask them about it, and the citizens are going to have website, mobile application, and computer application to just make their requests, and even without going to the government building unless they want to do just two things taking photos to process their service application or receive their services. And of course, we are going to provide online payment to make it easier to anyone.

1.3. Technologies used

- HTML, CSS, andJavascript for the Front-end
- Node.js on the backend
- Express.js for server and creating RESTful API
- SQL for database
- Java and Android Studio for mobile version
- Python for desktop version

1.4. Intended audience

This document is useful to: -

Customer: To follow up with the project team if there are requirements change.
Developer: To have ease of implementation.
Software Tester: To be able to develop the correct test cases for the system and test it to improve the system's quality.

Software architect: To be able to design and improve the system's architecture.
Project Manager: To be able to develop a good plan to work on the project and construct the team.

1.5. References

1.6. Overview

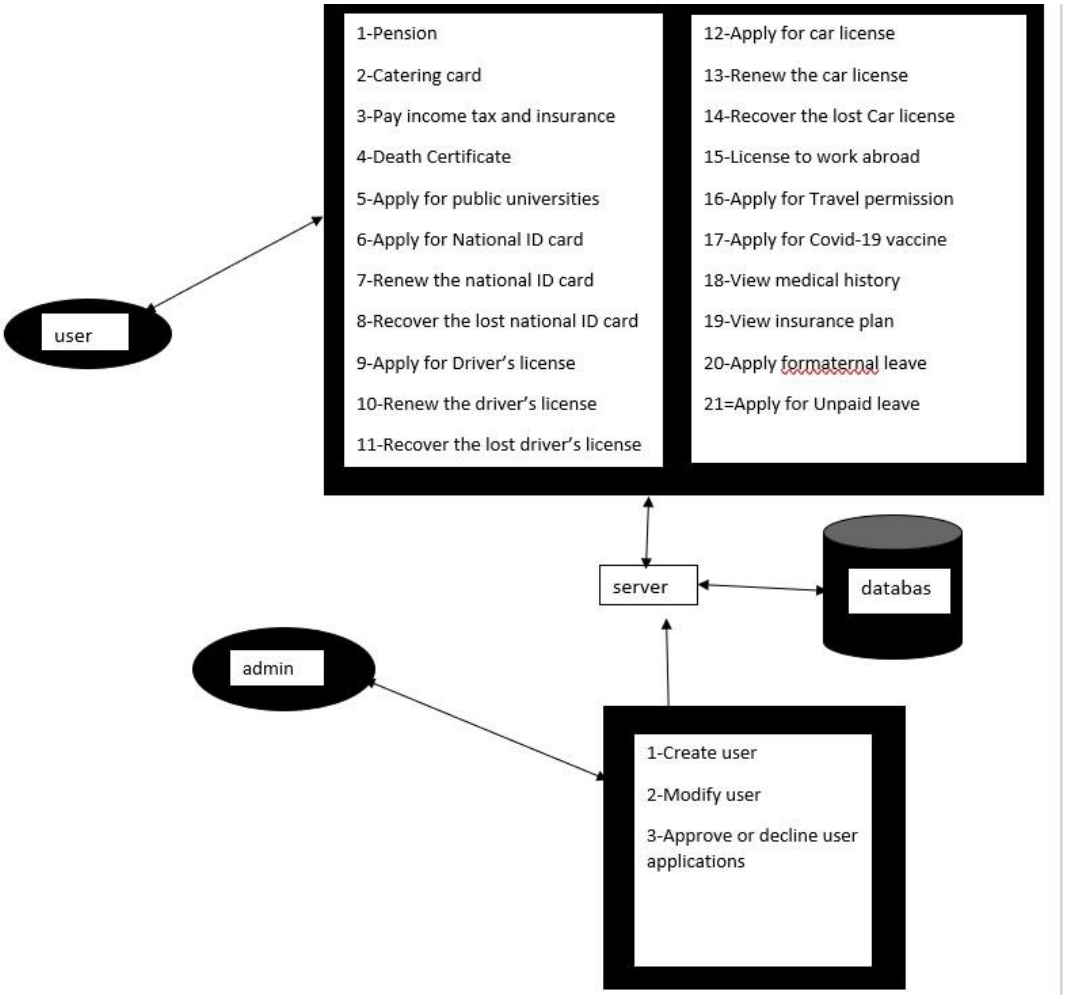
The rest of the SRS explains the detailed specifications of the people's Application. Chapter II of the SRS presents the general factors that affect the people's Application and its requirements, such as project constraints and user characteristics. Chapter III outlines the details of interfaces, and then the rest of the chapters' present the functional and nonfunctional requirements, Diagrams, and other related requirements of " People's app" Application.

"*People's app from the people to the people*"

2. Overall Description

2.1. Product Perspective

- People's app is a simple and intuitive app that will make it easier for people to use government services without needing to physically go to government buildings.
- The system will have two types of users: The user and the admin. Each has different functions and purpose.
- The system will use a centralized database which contains citizen data which will eliminate the need to go to different government institutions.



2.2. User Characteristics

There are two types of users who will interact with the system: the users and the admins

- The user is any citizen which needs to make use of a government service, such as applying for a national ID card.
- The admin will create user data and approve/decline user applications based on the user data.

2.3. Operating Environment

- The system will run on any desktop running Windows or Linux, and will be compatible with mobile phones running Android or iOS.

2.3. Constraints

- The only constraint for the app is an internet connection is required, an internet connection is required for the user to be able to submit data.

2.4.Assumptions and Dependencies

- It is assumed that a user will have a device compatible with the application such as a desktop or modern mobile phone
- It is assumed that the user will have a reliable internet connection.

3. Interfaces

3.1. System Interface

The first screen the user will see is the log-in page (Figure 2). The user will be able to log-in using their credentials.

After logging in, user will be able to see the services available to them. (Figure 3)

The user will be able to click on any one of the available services and view the required data, and will be able to submit them.

Figure 1 loading Screen

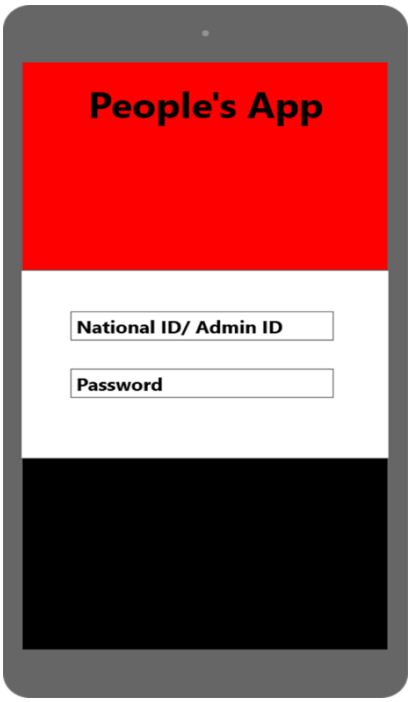


Figure 2 log in page

After Logging in, the user will be able to see the available services (Figure 3). After selecting a service, the user will see a sub-menu of the available options (Figure 4)

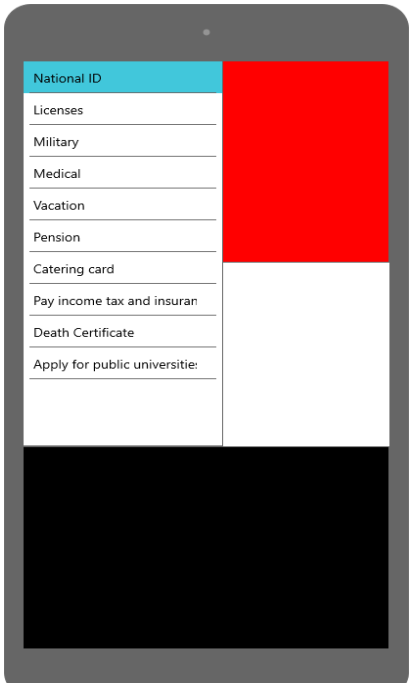


Figure3 Available services

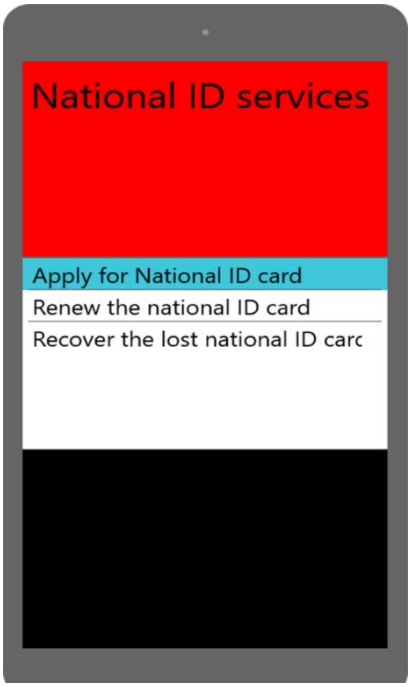


Figure4 Service sub-menu

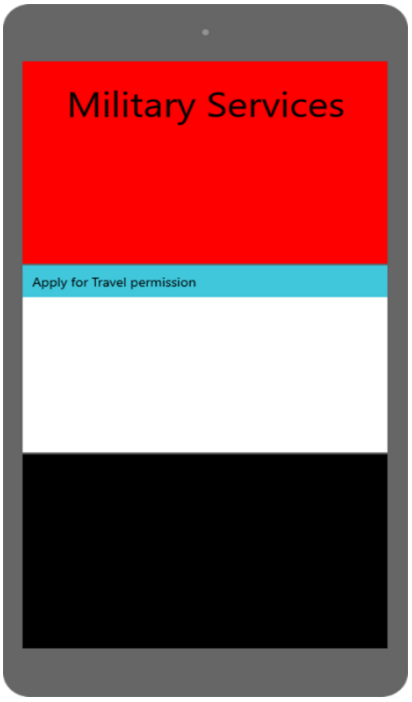


Figure 5

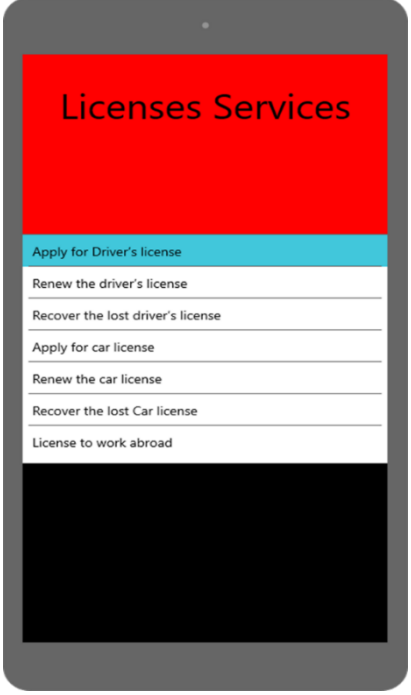


Figure 6

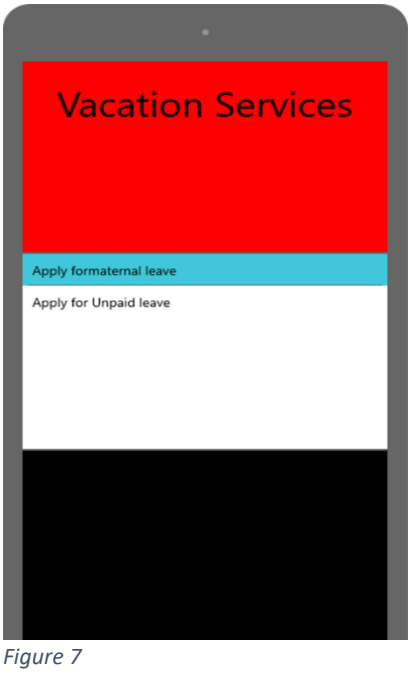


Figure 7

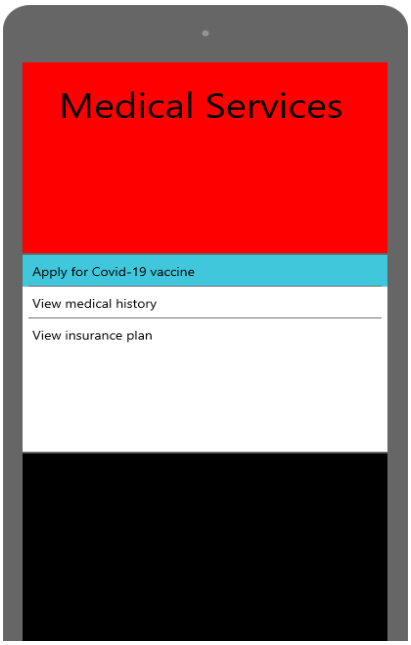


Figure 8

The admin can create/modify users and accept/decline user applications (Figure 9, 10)

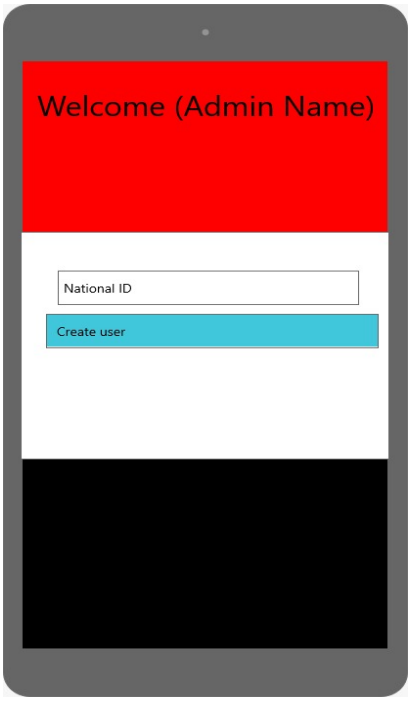


Figure 10

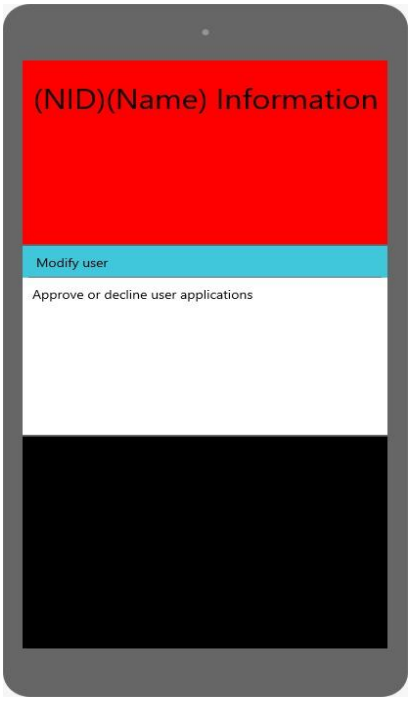


Figure 9

3.2. Software Interface

People's app is a cross-platform desktop application developed with Python and mySQL, external Python libraries will be used for User Interface and application will interact with a mySQL database, an HTTPS server will be used to handle requests from and to the application.

3.3. Hardware Interface

People's app is a desktop app that requires an internet connection for the user to be able to submit and access data, a camera is also required to be able to submit any needed photographs.

4. Functional Requirements

4.1 User Class 1 - The User

4.1.1 Functional requirements

Title: User Login.

Desc: System shall allow users to be able to enter their account by entering their personal details (National ID ,Password)

4.1.2 Functional requirements

Title: Apply for National ID card

Desc: System shall allow to apply to national ID card if they are above 16

4.1.3 Functional requirements

Title: Renew the national ID card

Desc: system shall allow the user to renew their national ID card

4.1.4 Functional requirements

Title: Recover the lost national ID card

Desc: System shall allow the user to recover their lost ID card

4.1.5 Functional requirements

Title: Apply for Driver's license

Desc: System shall allow to apply to driving license card if they are above 18

4.1.6 Functional requirements

Title: Renew the driver's license

Desc: System shall allow the user to renew the driver's license

4.1.7Functional requirements

Title: Recover the lost driver's license

Desc: System shall allow the user to recover the lost driver's license

4.1.8 Functional requirements

Title: Apply for car license

Desc: System shall allow the user to apply for car license

4.1.9 Functional requirements

Title: Renew the car license

Desc: System shall allow the user to renew the car license

4.1.10 Functional requirements

Title: Recover the lost Car license

Desc: System shall allow the user to recover the lost car license

4.1.11 Functional requirements

Title: Apply for Covid-19 vaccine

Desc: System shall allow user to apply for covid vaccine

4.1.12 Functional requirements

Title: Apply for Travel permission

Desc: System shall allow the male user to travel if they are in education and above 18

4.1.13 Functional requirements

Title: License to work abroad

Desc: System shall allow the user to apply for work abroad

4.1.14 Functional requirements

Title: Apply for maternal leave

Desc: System shall allow female user to take a 6 year vacation after giving birth till the baby become 18 (Birth Certificate)

4.1.15 Functional requirements

Title: Apply for Unpaid leave

Desc: System shall allow the user to apply for unpaid vacation

4.1.16 Functional requirements

Title: Apply for pension

Desc: System shall allow user to apply for pension if user is eligible for retirement

4.1.17 Functional requirements

Title: Apply for catering card (تموين)

Desc: System shall allow user to apply for catering card if user is a household caretaker and eligible

4.1.18 Functional requirements

Title: Pay income tax and insurance

Desc: System shall allow user to pay their income taxes and insurance fees via online payment

4.1.19 Functional requirements

Title: Death Certificate

Desc: System shall send user's death certificate to their relatives after user's death

4.1.20 Functional requirements

Title: Apply for public universities

Desc: System shall allow user to apply for public universities if they have graduated from high school

4.1.21 Functional requirements

Title: View medical history

Desc: System will allow user to view their medical history based on hospital records

4.1.22 Functional requirements

Title: View insurance plan

Desc: System will allow user to view their insurance plan and view hospitals which cover their insurance plan

4.2. User Class 2 - The Admin

4.2.1 Functional requirements

Title: Login as admin

Desc: Admins shall be able to log in as an admin by using their admin credentials

4.2.2 Functional requirements

Title: Create user

Desc: Admin can create user after user's birth by entering user's data (Full name, Parents' names, national ID, date of birth, location of birth, etc.)

4.2.3 Functional requirements

Title: Modify user

Desc: Admin can modify user's data

4.2.4 Functional requirements

Title: Approve or decline user applications

Desc: Admin can approve or decline a user's application based on the user data.

5. Non-Functional requirements

There are requirements for the system's performance, reliability, security, and user experience. They are as follows:

5.1 Reliability

- System should not fail unless there is a case of country wide power outage, as system should rely on backup servers in case of main server failure.

5.2 Recoverability

- System should not take more than 30 mins to run on backup servers in case of main server failure.

5.3 Performance

- System response time should not be over 2 seconds

5.4 Availability

- System should be running 24 hours a day, 7 days a week

5.5 Usability

- System should be intuitive and easy to use, user should not take more than 30 mins to be able to use system.

- System admin should receive 6 hours training time in order to be able to manage system adequately.

5.6 Maintainability

- System architecture is based on OOP which will make it easier to add or modify functionality in the future

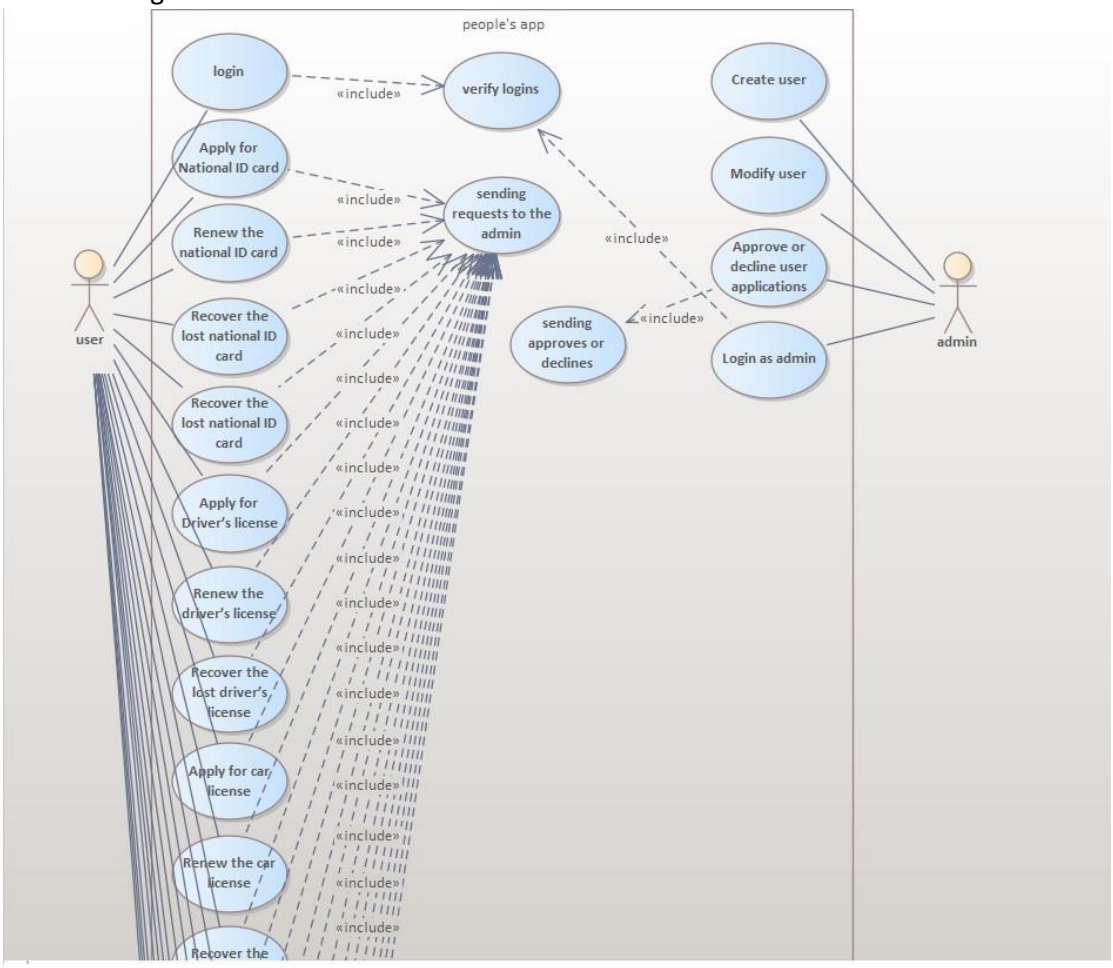
5.7 Security

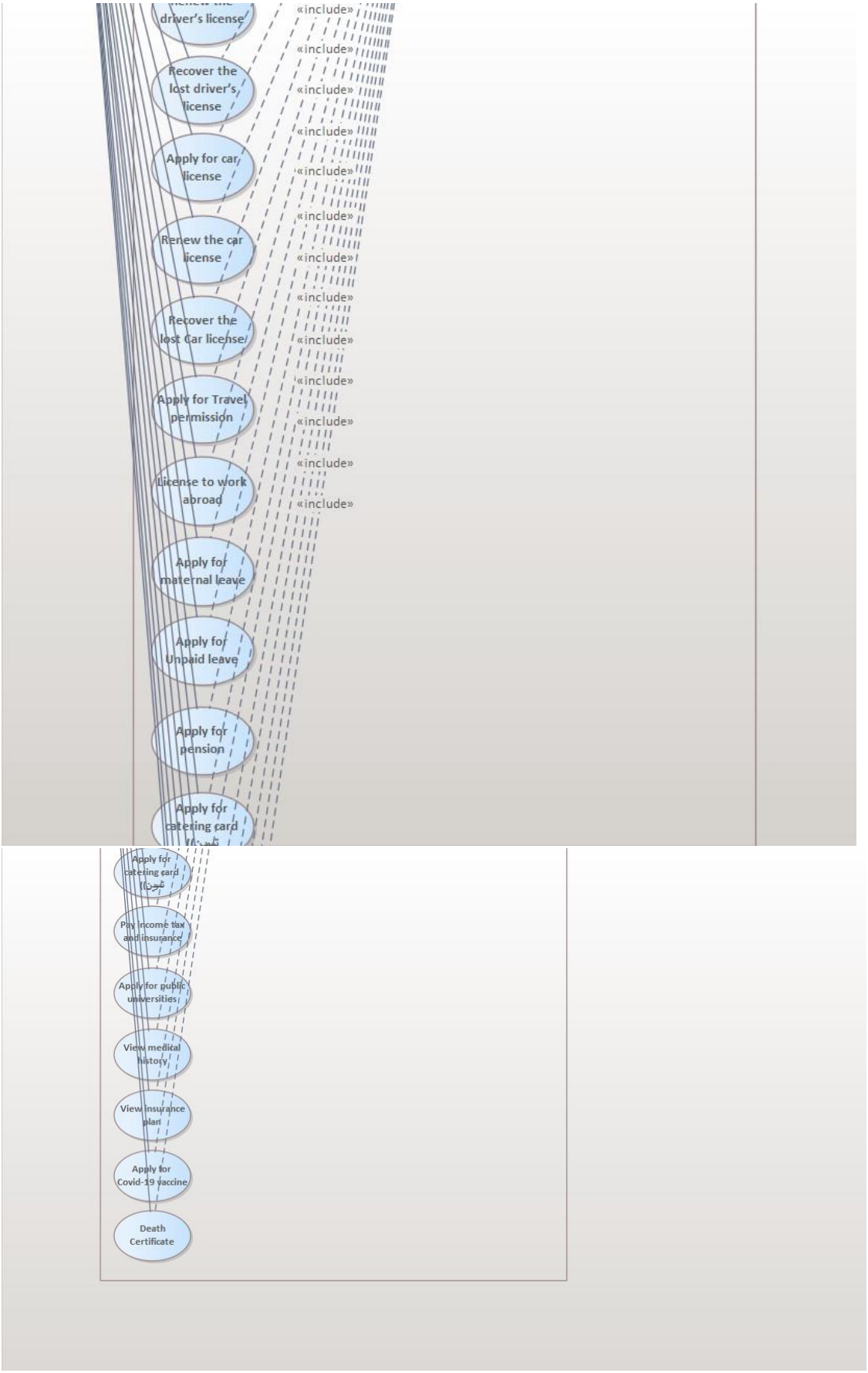
Since the system manages sensitive user data, security measures need to be in place in order to protect user data from being stolen or modified.

- Passwords will be encrypted by SHA-3method
- SQL Injection prevention methods will be in place, such as input validation.
- Secure Database privileges: Developers and admins can only read and write from the database, they cannot delete or modify table structure.
- Server will use HTTPS protocol for higher security.

6. Diagrams

Use case diagram




driver's license
Recover the lost driver's license
Apply for car license
Renew the car license
Recover the lost Car license
Apply for Travel permission
License to work abroad
Apply for maternal leave
Apply for Unpaid leave
Apply for pension
Apply for catering card
«include»
Apply for catering card
Pay income tax and insurance
Apply for public universities
View medical history
View insurance plan
Apply for Covid-19 vaccine
Death Certificate

Use case scenarios

Use case 1

Usecase name	Sign-in
Actors	User/Admin
Main success scenario	1- System prompts user for username and password 2- User enters required information 3- User clicks on sign-in 4- System checks the username and password entered 5- System signs the user in
Exceptions	3a User clicks on sign in without completing the required fields
Actions	3.1 System displays error message “The marked fields are required” 3.2 User fills in the missing information
Pre-condition	User has downloaded the application User has an active internet connection
Post-condition	User is successfully signed in

Use case 2

Usecase name	Create User
Actors	Admin
Main success scenario	1- System prompts admin for user’s data (Full name, Parents’ names, national ID, date of birth, location of birth, etc.) 2- Admin enters required information 3- Admin clicks on create user 4- System checks that user doesn’t already exist and data is valid 5- System creates the user
Exceptions	1a Admin clicks on create user without completing all the required fields 4a Admin enters a user that already exists
Actions	1.1 System displays error message “The marked fields are required” 1.2 admin fills in the missing information 4.1 System displays error message “User already exists”

Pre-condition	User does not already exist
Post-condition	User is successfully created

Use case 3

Usecase name	**Modify User**
Actors	Admin
Main success scenario	1- System displays user's info 2- Admin modifies any of the info 3- Admin clicks on save changes 4- System checks that input is valid and no fields are empty 5- System updates the user
Exceptions	1a Admin clicks on save changes while one field or more is empty
Actions	1.1 System displays error message "The marked fields are required" 1.2 admin fills in the missing information
Pre-condition	No user data is missing
Post-condition	User is successfully modified

Use case 4

Usecase name	**Approve or decline user applications**
Actors	Admin
Main success scenario	1- System displays pending user applications 2- Admin selects application to review 3- Admin clicks on approve or decline 4- System sends the approved application to the ministry and notifies the user
Exceptions	3a Admin's internet connection is faulty

Actions	3.1 System displays error “Unable to communicate with server, please check internet connection.”
Pre-condition	Application has been submitted by user
Post-condition	Application is successfully approved and is sent to the appropriate ministry

Use case 5

Usecase name	**Apply for ID card**
Actors	User
Main success scenario	1- System prompts user for required information 2- User enters required information 3- User clicks on apply 4- System checks that user is above 16 years old, all data has been filled, and user does not already have an ID card 5- System sends the application to be reviewed by admin
Exceptions	3a User clicks on apply without completing all the required fields 4a User is not 16 or older 4b User already has an ID card
Actions	3.1 System displays error message “The marked fields are required” 3.2 user fills in the missing information 4.1 System displays error message “User must be 16 or older” or “User already has an ID card”
Pre-condition	User is 16 or older and has not already applied for ID
Post-condition	Application is successfully sent to be reviewed by admin

Use case 6

Usecase name	**Apply for Driver’s License**
Actors	User

Main success scenario	1- System prompts user for required information 2- User enters required information 3- User clicks on apply 4- System checks that user is above 18 years old, all data has been filled, and user does not already have driver's license 5- System sends the application to be reviewed by admin
Exceptions	3a User clicks on apply without completing all the required fields 4a User is not 18 or older
Actions	3.1 System displays error message "The marked fields are required" 3.2 user fills in the missing information 4.1 System displays error message "User must be 18 or older" or "User already has a license"
Pre-condition	User is 18 or older and has not already applied for a driver's license
Post-condition	Application is successfully sent to be reviewed by admin

Use case 7

Usecase name	**Apply for Car License**
Actors	User
Main success scenario	1- System prompts user for required information 2- User enters required information 3- User clicks on apply 4- System checks that user is above 18 years old, all data has been filled, and user does not already have Car License 5- System sends the application to be reviewed by admin
Exceptions	3a User clicks on apply without completing all the required fields 4a User is not 18 or older
Actions	3.1 System displays error message "The marked fields are required" 3.2 user fills in the missing information

	4.1 System displays error message “User must be 18 or older” or “User already has a Car License ”
Pre-condition	User is 18 or older and has not already applied for a Car License
Post-condition	Application is successfully sent to be reviewed by admin

Use case 8

Usecase name	**Apply for Covid Vaccine**
Actors	User
Main success scenario	1- System prompts user for required information 2- User enters required information 3- User clicks on apply 4- System checks that all data has been filled, and user has not already applied 5- System sends the application to be reviewed by admin
Exceptions	3a User clicks on apply without completing all the required fields 4a User has already applied
Actions	3.1 System displays error message “The marked fields are required” 3.2 user fills in the missing information 4.1 System displays error message “User has already applied ”
Pre-condition	User has not already applied for the vaccine
Post-condition	Application is successfully sent to be reviewed by admin

Use case 9

Usecase name	**View medical history**
Actors	User

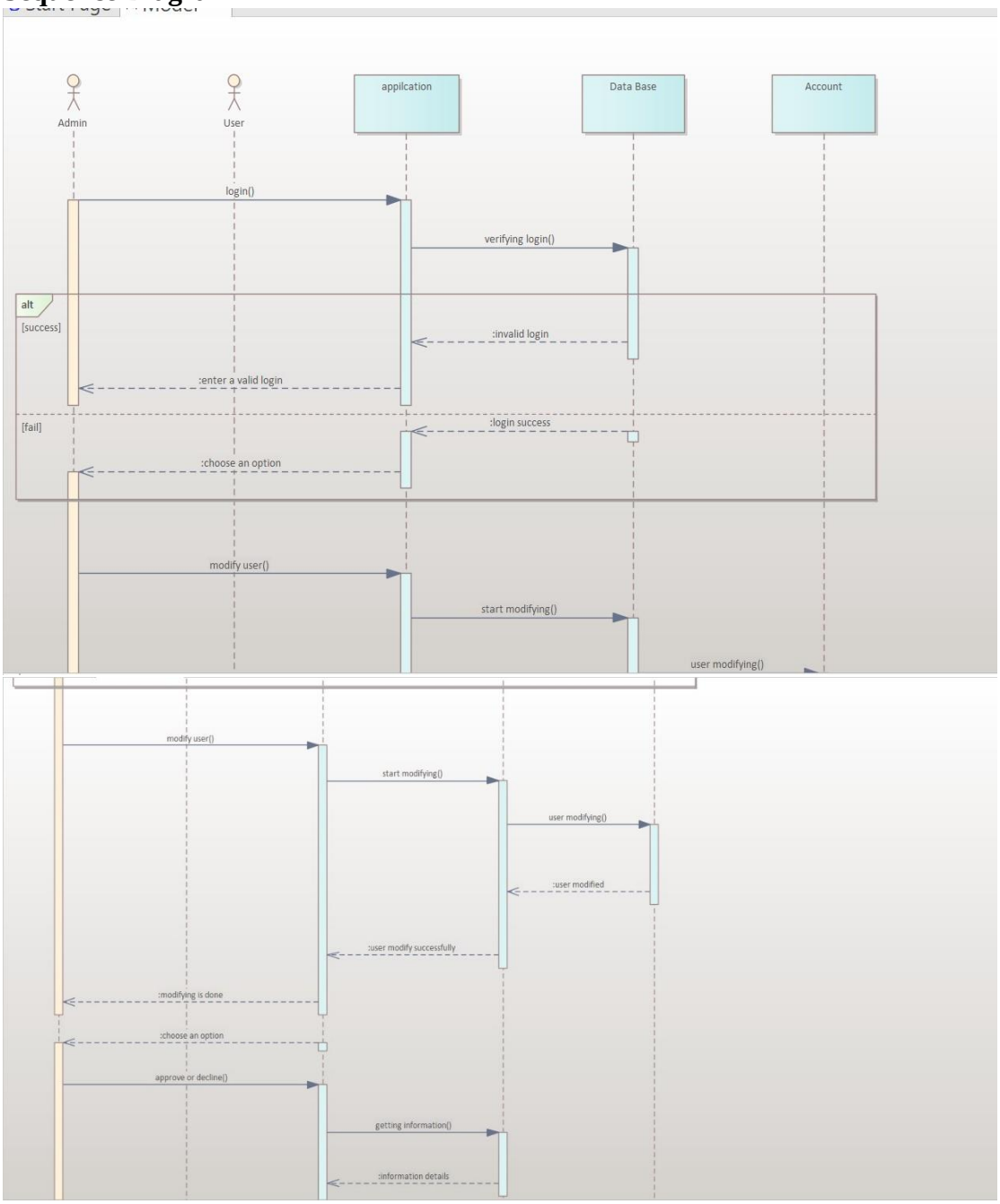
Main success scenario	1- User clicks on “View medical history” 2- System displays user’s hospital records
Exceptions	2a User has no medical records
Actions	2.1 System displays message “No medical history found”
Pre-condition	User is logged in
Post-condition	User’s medical history is displayed

Use case 10

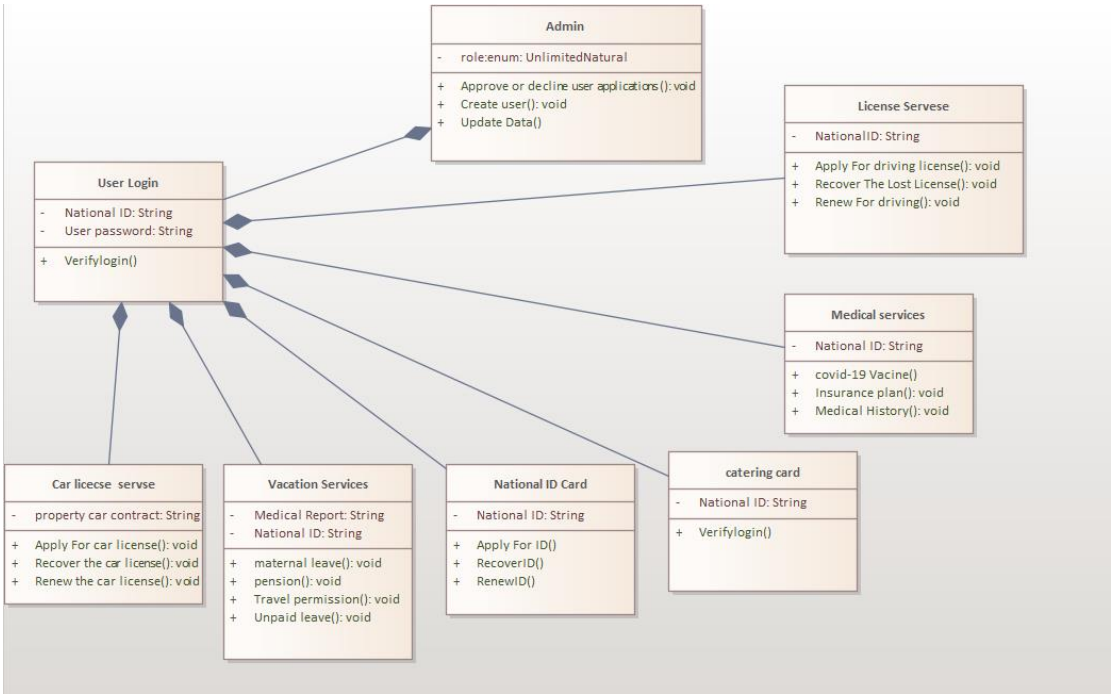
Usecase name	**Apply for pension**
Actors	User
Main success scenario	1- System prompts user for required information 2- User enters required information 3- User clicks on apply 4- System checks that all data has been filled, user has not already applied, and user is eligible for pension 5- System sends the application to be reviewed by admin
Exceptions	3a User clicks on apply without completing all the required fields 4a User has already applied 4b User is not eligible for pension
Actions	3.1 System displays error message “The marked fields are required” 3.2 user fills in the missing information 4.1 System displays error message “User has already applied ” 4.2 System displays error message “User is not eligible for pension”
Pre-condition	User has not already applied for the pension User is eligible

Post-condition	Application is successfully sent to be reviewed by admin

Sequence Diagram



Class diagram



https://nileuniversity-my.sharepoint.com/:v:/g/personal/sal_ayman_nu_edu_eg/EQhffWxfY1JMqieRVPglOKIBjt5Yl_NaKzp6z7gCJlH9Bw?e=kJWcnn