12/6/2024

(Student C00270632) Qadeer Hussain Supervisor: Paul Barry

Software DEVELOPMENT PROJECT YEAR 4

Design Document

Elderly Care Management System(ECMS)

A logo with black text

Description automatically generated

**Table of Contents**

[**Introduction** 2](#_Toc184322226)

[**System Architecture** 3](#_Toc184322227)

[**Technologies & Tools** 4](#_Toc184322228)

[**System Sequence Diagrams** 5](#_Toc184322229)

[**Patient Profile** 5](#_Toc184322230)

[**Prototype GUI Screens** 6](#_Toc184322231)

[**Main Menu** 6](#_Toc184322232)

[**Patient Profile Prototype** 6](#_Toc184322233)

[**Database Design** 7](#_Toc184322234)

[**Patient Profile Table** 7](#_Toc184322235)

[**Detailed Use Cases** 8](#_Toc184322236)

[**Patient Profile Use Case** 8](#_Toc184322237)

[**References** 10](#_Toc184322238)

**Figure Table**

[Figure 1: System Architecture 3](#_Toc184319000)

[Figure 2: Patient Profile Sequence Diagram 5](#_Toc184319001)

[Figure 3: Main Menu 6](#_Toc184319002)

[Figure 4: Patient Profile 6](#_Toc184319003)

[Figure 5: Patient Profile Table 7](#_Toc184319004)

[Figure 6: Example Patient Data 7](#_Toc184319005)

[Figure 7: Patient Profile ER diagram 7](#_Toc184319006)

# **Introduction**

The purpose of this design document is to outline key features of the Elderly Care Management System(ECMS). The document provides information relating to the design of the application.

* The System Architecture Diagram that outlines the overall structure of the application
* System Sequence Diagrams will be included in the document to illustrate the functionality of the features.
* GUI Prototype Screens.
* The Database Design to illustrate how the data is organised and managed .
* Detailed Use Case will be provided to demonstrate the internal functions of the features.

# **System Architecture**

A diagram of a computer

Description automatically generated

Figure 1: System Architecture

This system architecture diagram for the Elderly Care Management System illustrates the overall structure of the project. This diagram contains the database that stores all data which is safe guarded by encryption and decryption. The main interface accessible by both Admin and Carer.

# **Technologies & Tools**

This project uses multiple “technologies and tools”. This section of the document outlines the technologies and tools used: Jupyter Notebook, Cryptography Library, Maria DB, Python and Django.

**Jupyter Notebook**

Jupyter Notebook is utilized for prototyping and data exploration of the early phases of the project. This is particularly useful for testing database queries, experimenting with data encryption methods, and visualizing data.

**Cryptography Library**

The cryptography library is crucial to ensure data security within the application. This library was used to encrypt data and decrypt data providing robust protection for sensitive information handled by the system.

**MariaDB**

MariaDB was selected as the database management system to store and manage data throughout the development of this project. This was chosen as it has been reliable in previous projects.

**Python**

Python will be the core language used in this project, helping with backend tasks, handling data, and connecting different parts of this system. It has many libraries that make it easy to work.

**Django**

Django is a high level python web framework. It will be used throughout the project, helping to enhance security and streamlining the development of the application.

# **System Sequence Diagrams**

## **Patient Profile**

A diagram of a patient profile

Description automatically generated

Figure 2: Patient Profile Sequence Diagram

In the Elderly Care Management System their is a key feature which is the Patient Profile. This keeps a secure record of the patient personal and medical information.

# **Prototype GUI Screens**

## **Main Menu**

A blue background with orange labels

Description automatically generated

Figure 3: Main Menu

## **Patient Profile Prototype**

A screenshot of a contact form

Description automatically generated

Figure 4: Patient Profile

# **Database Design**

## **Patient Profile Table**

A screenshot of a computer screen

Description automatically generated

Figure 5: Patient Profile Table

**Example Patient Data**

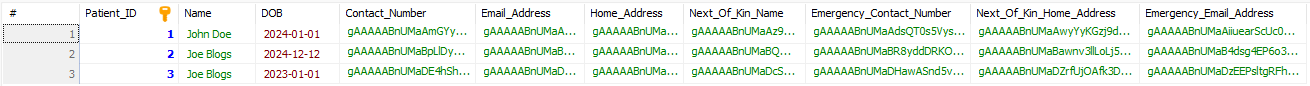
****

Figure 6: Example Patient Data

**Patient Profile ER diagram**

A screen shot of a computer

Description automatically generated

Figure 7: Patient Profile ER diagram

# **Detailed Use Cases**

## **Patient Profile Use Case**

|  |
| --- |
| **Use Case Name:** Create Patient |
| **Actors:** Admin |
| **Pre-Conditions:** The user must be logged into the ECMS system. |
| **Main Success Scenario:**   1. The user logged into the system. 2. The user selects patient profile. 3. The user selects Add new patient. 4. Input the patient information. |
| **Post-Conditions:** The Patient Profile is created |
| **Alternative Scenario:** Incorrect input or invalid data prompt user to correct the data |

|  |
| --- |
| **Use Case Name:** Encrypt |
| **Actors:** Admin, System |
| **Pre-Conditions:** Patient data is valid and is ready for encryption. |
| **Main Success Scenario:**   1. The System gets information from the user. 2. The System encrypts the sensitive data fields. 3. The encrypted patient data is ready to be stored |
| **Post-Conditions:** The data is encrypted. |
| **Alternative Scenario:** Patient data fails to encrypt it notify the user. |

|  |
| --- |
| **Use Case Name:** Store |
| **Actors:** System |
| **Pre-Conditions:** Database connection is established |
| **Main Success Scenario:**   1. The encrypted patient data sent to database. 2. The Data is stored to relevant table and column. 3. The system logs the storage. |
| **Post-Conditions:** The patient data is successfully stored in the database. |
| **Alternative Scenario:** Patient data fails to be stored in the database this notifies thee user. |

|  |
| --- |
| **Use Case Name:** Search/Find |
| **Actors:** Admin, Carer |
| **Pre-Conditions:** The database contains patient data to search. |
| **Main Success Scenario:**   1. The user is logged into the system 2. The user selects patient profile. 3. Search function. 4. System retrieves relevant record. |
| **Post-Conditions:** Successfully find patient profile. |
| **Alternative Scenario:** If no Patient profile is found, the system informs the user that no results are available. |

|  |
| --- |
| **Use Case Name:** Retrieve/Display |
| **Actors:** Admin, Carer, System |
| **Pre-Conditions:** The data exists in the database and is accessible for viewing, |
| **Main Success Scenario:**   1. The user selects patient record. 2. System decrypts the patient data. 3. The system then displays the patients data. |
| **Post-Conditions:** The patient’s data is displayed to the Admin or Carer. |
| **Alternative Scenario:** If patient is unable to display and/or decrypt patient data notify the user. |

# **References**

Django Software Foundation, 2024. *Django.* [Online]   
Available at: https://www.djangoproject.com/  
[Accessed December 2024].

Jupyter, 2024. *Jupyter.* [Online]   
Available at: https://jupyter.org/  
[Accessed December 2024].

MariaDB Foundation , 2024. *MariaDB Server.* [Online]   
Available at: https://mariadb.org/  
[Accessed December 2024].

PYPI, 2024. *Cryptography.* [Online]   
Available at: https://pypi.org/project/cryptography/  
[Accessed December 2024].

Python Software Foundation, 2024. *Python.* [Online]   
Available at: https://www.python.org/  
[Accessed December 2024].