A logo with text on it

Description automatically generated

RIGA TECHNICAL UNIVERSITY

Science and Information Technology Faculty

**Institute of Applied Computer Systems**

**Health Clinic Appointment Management System**

Mohammed Arham Mohammed 221ADB053

Sadeem Riyaz 221ADB072

Furqan Ul Islam 211ADB086

Software Design Description

Version 2.0

Riga 2024

**Contents**

Health Clinic Appointment Management System

1. INTRODUCTION.
   1. Definition of abbreviations.
   2. Document purpose.
   3. Scope.
   4. Related Documents.
   5. Document overview.
2. CLIENT USER INTERFACE.
   1. Authorization form.
   2. Patient information.
   3. Doctor’s dashboard.
3. DESCRIPTION OF THE DECOMPOSITION.
   1. Prototype database ER model.
   2. Description of the database.
4. SOFTWARE SUMMARY.
5. **INTRODUCTION**
   1. Definition of abbreviations

|  |  |
| --- | --- |
| Abbreviations | Explanation |
| HCAMS | Health Clinic Appointment Management System |
| DBMS | Database management system |
| GUI | Graphical User Interface |
| OS | Operating System |

* 1. Document purpose.

This document describes the software requirements of the projects “Health Clinic Appointment Management System.”

The document is intended for the parties involved in the development, implementation, and maintenance of the software development within the study project:

* Receptionists, who are responsible for creating profiles of verified clients(patients).
* Doctors, who are responsible for reviewing client(patients) information created by receptionists.
* Technical specialists of the developer, who are responsible for its implementation - design and implementation.
  1. Scope

The software “Health Clinic Appointment Management System.” creates a system for doctors with clinics to schedule and keep track of the patient's appointments.

This document describes how the users will operate the software:

* Requires receptionist to verify the client(patient)
* The receptionist takes the relevant information and creates a client profile.
* Receptionists identifies the clients’ needs and assigns them an appointment accordingly.
* The doctor gets notified about the appointment scheduled.
* After the appointment, the doctor sends the prescription to the receptionists.
* The client’s visits and prescription that is given is stored in a database.
  1. Related Documents.

This software design document is made from referencing the following document:

[[Example] System Design Description](https://estudijas.rtu.lv/mod/resource/view.php?id=3947805). *(From ortus)*

* 1. Document overview.

This document consists of four sections:

* The first section - Introduction, contains information about the general structure, purpose and definitions used.
* In the second section - Description of the Solution Process, the main steps of the process are described.
* The third section, Specific Requirements, describes all the solution requirements that apply to this software.
* In the fourth section - General limitations of the solution.

**2. CLIENT USER INTERFACE**

**2.1. Authorization form**

2.1.1. Input parameters.

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD NAME** | **FIELD TYPE** | **OBLIGATORY** | **DESCRIPTION** |
| Username | Text input field | Yes | User authorization Username |
| Password | Text Input field | Yes | User authorization Password |

2.1.2. Function description

1. By pressing the “Login” button the user authorization data is checked with the data in the database. Incase of incorrect input data, a corresponding message is displayed or, if the input data is correct, the relevant data is loaded. If the user with the role receptionists logs into the system, the form “Patient information” is opened, or if the user with the role doctor logs into the system, option is given to choose “Schedule”, “Patient information” and “Feedback”.

2.1.3. Screen Prototype.

A screenshot of a login screen

Description automatically generated

**2.2” Patient information”**

2.2.1. Input parameters.

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD NAME** | **FIELD TYPE** | **OBLIGATORY** | **DESCRIPTION** |
| Name | Text Input Field | Yes | Client name |
| Surname | Text Input Field | No | Client surname |
| Gender | Dropdown option | Yes | Client gender |
| Date of Birth | Date | Yes | Client date of birth |
| Height (cm) | Number Input Field | Yes | Client height |
| Weight(kg) | Number Input Field | Yes | Client weight |
| Email | Text Input Field | Yes | Client email |
| Phone Number | Number Input Field | Yes | Client number |
| Insurance | Checkbox | No | Client insurance |
| Insurance Number | Number Input Field | No | Client insurance number |

2.2.2. Function description

1. The user with the role receptionists will have to fill up the information about the patient.
2. By pressing the “Yes” checkbox on the bottom next to “Insurance” a textbox to enter the insurance number.
3. After the relevant data is entered by the user, “Submit” button should be pressed.
4. Pressing the “Submit” button will enter the data into the database.

2.2.3. Screen prototype.

A screenshot of a patient information form

Description automatically generated

**2.3. “Doctor’s Dashboard”**

2.3.1. Input Parameters.

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD NAME** | **FIELD TYPE** | **OBLIGATORY** | **DESCRIPTION** |
| Prescription | Text input field | No | Prescription provided by doctor |

2.3.2. Function description.

1. The user with the role of doctor is given a text input box to add a prescription given to the relevant patient.

2.3.3. Screen prototype,

*A screenshot of a medical form

Description automatically generated*

**3.DESCRIPTION OF THE DECOMPOSITION**

3.1. Prototype database ER model.

A screenshot of a computer program

Description automatically generated

3.2. Description of the database

3.2.1. R\_users

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| IDENTIFICATOR | | R\_users | | |
| DESCRIPTION | | | | |
| The table that stores receptionist’s user data | | | | |
| **FIELD NAME** | **FIELD TYPE** | | **OBLIGATORY** | **DESCRIPTION** |
| R\_username | Varchar (100) | | Yes | Receptionist’s username |
| R\_password | Varchar (100) | | Yes | Receptionist’s password |

3.2.2. D\_users

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| IDENTIFICATOR | | D\_users | | |
| DESCRIPTION | | | | |
| The table that stores Doctor’s user data | | | | |
| **FIELD NAME** | **FIELD TYPE** | | **OBLIGATORY** | **DESCRIPTION** |
| D\_username | Varchar (100) | | Yes | Doctor’s username |
| D\_password | Varchar (100) | | Yes | Doctor’s password |

3.2.2. Patient\_data

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| IDENTIFICATOR | | Patient\_data | | |
| DESCRIPTION | | | | |
| This table stores patient’s data | | | | |
| **FIELD NAME** | **FIELD TYPE** | | **OBLIGATORY** | **DESCRIPTION** |
| P\_id | Varchar (100) | | Yes | Patient ID |
| P\_Name | Varchar (100) | | Yes | Patient name |
| P\_Surname | Varchar (100) | | No | Patient surname |
| P\_Gender | Boolean | | Yes | Patient gender |
| P\_DOB | Date | | Yes | Patient date of birth |
| P\_Height (cm) | Decimal (3,2) | | Yes | Patient height |
| P\_Weight(kg) | Decimal (3,2) | | Yes | Patient weight |
| P\_Email | Varchar (100) | | Yes | Patient email |
| P\_Phone | Varchar (100) | | Yes | Patient number |
| P\_Insurance | Boolean | | No | Patient insurance |
| P\_Insurance\_number | Varchar (100) | | No | Patient insurance number |
| P\_Prescription | Text | | No | Patient Prescription |

**4.SOFTWARE SUMMARY**

1. When the software is launched the users get an option to specify who they are (receptionists or doctor) and a login page (2.1.3) to get access. The user logs in by entering their username and password. The user’s username and password are added to the database by the developer manually.
2. There are two types of users, i.e. a receptionist and a doctor. If the user is a receptionist, a screen of input fields for adding patient information is loaded (2.2.3). If the user is a doctor, a screen of their schedule and patient information is loaded (2.3.3) with an input field to add prescription.
3. The patient information along the prescription given is all stored in one table in the DBMS.