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 Test Report

Version 1.0

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 identify 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] Test Report](https://estudijas.rtu.lv/mod/resource/view.php?id=3947807). *(From ortus)*

* 1. General requirements for test execution

In order to test operation of the system user needs:

* Access to the system
* User name and password
* User rights to the Administrators role.

**2. Function Testing**

|  |  |  |
| --- | --- | --- |
| **No.** | **DESCRIPTION** | **PPS REQUIREMENT** |
| 1. | User Authorization | HCAMS-1 |
| 2. | User Administration | HCAMS-2 |
| 3. | Test Administration | HCAMS-3 |
| 4. | Test Execution | HCAMS-4 |
| 5. | Reporting | HCAMS-5 |

**3. Test Scenarios**

3.1. User Authorization.

Open the authorization form.

3.1.1. Checking if the required fields are filled:

* Enter information in fields.
* Press the "Login" button.

Expected result: A message is displayed that no mandatory fields have been entered.

3.1.2. Checking the username and password.

* Enter the value “test” in the “Username” field.
* Enter the value “test” in the “Password” field.
* Press the “Login” button.

Expected result: A message is displayed that an incorrect username or password has been entered.

3.1.3. Successful user authorization.

* Enter the correct username in the “Username” field.
* Field Enter the correct password in the “Password” field.
* Press the “Login” button.

Expected result: The user is successfully authorized.

3.2. User administration.

Authorize with a user who has the role "Receptionists". It should open receptionist’s dashboard.

3.2.1. Display the user list data.

* Press the “Patients” button to view patient data.
* Press the “Doctor” button to view Doctors data.

3.2.2. Filtering data of the patients

* Enter the value “non-existent” in the “Name” field.
* Press the “Search” button.

Expected result: An empty list of users is displayed.

3.2.3. To edit user data

* In the list, select the user and click on the entry edit icon.
* We change the username in the "Name" field.
* Click the "Save" button.

3.2.4. To create Session

* Click “Add a session” button.
* Fill in the relevant details.

Expected results: A session is created. Any registered patient can book an appointment.

3.2.5. To delete an appointment.

* Click “Delete”.

Expected results: The appointment booked by a patient will be deleted.

3.3 Doctor Dashboard.

User with the role of doctor can do the following operations:

* View appointment.
* Cancel appointment.
* View session.
* Cancel session.
* View patient information.
* Modify account details.

3.3.1. Display appointment.

Expected results: Doctor’s Appointment list is displayed.

3.3.2. Canceling appointment.

* Click “Cancel” button under events.

Expected results: Appointments booked by patients should be removed from appointment list.

3.3.3. Display session.

* Click “View” button under events.

Expected results: Session list is displayed.

3.3.2. Canceling Session.

* Click “Cancel” button under events.

Expected results: Session created by receptionists should be removed from the sessions list.

3.3.4. Display patient information.

* Click “View” button under events.

Expected results: Patient information is displayed.

3.4.5. Modify Account details.

* Click “Delete Account” button in the settings section.

Expected results: Doctors account should be deleted.