<Online Hospital System>

System Design Document

<0.1>

<Date>

<Selin Atalay

Hasan Cihan Akpınar

Murat Akalın

Tolga Şıvgın>

Prepared for

SE301 Software Engineering



Table of Contents

[1. Introduction 1](#_Toc433996772)

[1.1. Purpose of the System 1](#_Toc433996773)

[1.2. Design Goals 1](#_Toc433996774)

[1.3. Definitions, Acronyms, and Abbreviations 1](#_Toc433996775)

[1.4. References 1](#_Toc433996776)

[2. Current Software Architecture 1](#_Toc433996777)

[3. Proposed Software Architecture 1](#_Toc433996778)

[3.1. Overview 1](#_Toc433996779)

[3.2. System Decomposition 1](#_Toc433996780)

[3.3. Hardware Software Mapping 2](#_Toc433996781)

[3.4. Persistent Data Management 2](#_Toc433996782)

[3.5. Access Control and Security 2](#_Toc433996783)

[3.6. Global Software Control 2](#_Toc433996784)

[3.7. Boundary Conditions 2](#_Toc433996785)

[4. Subsystem Services 2](#_Toc433996786)

[5. References 2](#_Toc433996787)

# Introduction

SDD document designed for any user.Therefore, system architecture and properties must be understandable.It summarizes software architecture and design goals.System design be defined to meet the requirements in the using system and provide the best services to users.We have divided the project into smaller units to implement with easy terms and best the Hospital Appointment system that we done.

In this Project firstly We identify the needs of present system.We designed the system that we will do in line with these requirements.

## Purpose of the System

Purpose of the Hospital appointment system devise a appointment system with

simple,easy to use, and internet and web browser that can be usable from any device

as described in the previous report of Requirements Specification and Analysis. Users can enter the system and easily make an appointment, see the test results and choose the outpatient clinic.

## Design Goals

When they enter the system, they can choose a polyclinic. They book an appointment or cancel an appointment if they want.At the same time all users view appointment at the part of appointments.They can see the test results.

**Security :** The system security is one of the most important non-functional requirements.

**Usability:** The system should be user friendly and easy to understand and use.

**Reliability:** The system will be safe.All user enter the system with their TC numbers and password.

**Flexibility:** It requires Internet and available web browser with the device.

**Implementation:** System should be modified and readable.The number of errors must be minimized.

**Interface:** The interface should be as good as possible.

**Supportability:** Developer is responsible for testabilty, modified and compatibility of the system.

**Scalability:** Application should be able to handle large numbers of clients.

**Performans:** The system has to be sturdy enough to manage any valid input from the users.

## Definitions, Acronyms, and Abbreviations

LFH: A name of the system

User :Who use the LFH system

Personnel: : A person who Setting appointments and update doctor information

Admin: : A person who do some operations on database.

RAD: Requirement Analysis Document.

## References

-There are many Hospital Appointment System on internet but our system different from others.

# Current Software Architecture

Our system is not an existing system. The hospital is new, so we will create a new system. When the patient(user) registers with the system and logs in, he/she can make an appointment with each department, get information about doctors and clinics and reach the lab results. When the personnel logs in to the system, they can edit or delete the user's appointments. Other than that, the doctor can add working hours and laboratory results. When the administrator login the system, he/she can add new personnel, delete personnel or edit the personnel information. Likewise, administrator can do these for doctors. And finally, to give this system a name, we call it the Life Hospital system.

# Proposed Software Architecture

Life Hospital System is a web-based system. This project for a new hospital and this system offers the possibility to choose according to status of user. Our system has a difference of user-friendly interface and reliable, than the like other hospital systems. If user is patient, login on system and he/she get an appointment from this system. Also this system offer opportunity to cancel appointment. Other user is Hospital Personnel which is setting available doctors times, adding laboratory results on system and editing patients’ appointments on system. Finally, users have knowledge about hospitals’ vision and mission on this website. According to the like other hospital systems, we tried to make our system easy to use, open and easy to develop.

## Overview

## I) User Mode

* **Register;**

Patients may register on the system with required information. If users not registered in the system, they can login the system after registered

* **Request Password Reminder;**

When patient forgot his/her password, use reminder password button to remember password.

* **Authentication;**

Patients can enter the system with own TC number and password which is created by own on register. Patients who is previously registered in the system can access next step from this part.

* **Book an Appointment;**

When patients enter get appointment option, they do selection hospital, department, doctor, day and hour step by step.

* **Save an appointment;**

After he or she checked appointment’s information on the page which is show patient’s choosing if all information are true he/she click the save button and get an appointment.

* **Cancel an appointment;**

Patient’s all appointment showing same page on system. Patient click the appointment button if patient is not going to go to appointment then he can cancel which he wants.

* **Logs Off;**

Patients exit the system when they finished their operations on the system.

**II) Personnel mode**

* **Authentication;**

Personnel part is like Admin Authenticationand Patient Authentication. In this authentication , personnel can login on system with TC number and password. Password is given by the Admin.

* **Edit appointment information;**

Personnel update information which must be change, on saved program by personnel.

* **Enter Lab Resut;**

Personnel record all Lab Result in system. Personnel do enter Patient information and enter Lab Result of Patient.

* **Edit Lab Result information;**

Personnel update information which must be change, on saved program by personnel.

* **Log Off;**

Personnel exit the system when they finished the operations on the system.

* **Set Available Date for Doctor ;**

First of all personnel choose hospital which he/she is work there. Then enter information (department, doctor, date and hour) which are contained monthly work program. Monthly program is given to him/her by doctors.

**III) Admin mode**

* **Authentication;**

Admin part on home page, admin can login on system with TC number and password.

* **List the Upcoming Doctor**

Admin sees doctor who working on the hospital.

* **Add Doctor;**

Admin updates which is necessary such as a doctor come to work the hospital.

* **Delete Doctor;**

Admin update which is necessary such as a doctor arrive the hospital.

* **Edit Doctor;**

Admin update doctor information such entered wrong information of doctor.

* **List the Personal**

Admin sees all personal.

* **Add Personal;**

Admin updates which is necessary such as a new personal come to work the hospital.

* **Delete Personal;**

Admin update which is necessary such as a personal arrive the hospital.

* **Edit Personal;**

Admin update personal information such entered wrong information of personal.

* **List the Policlinic;**

Admin sees all policlinic

* **Add Policlinic;**

Admin can add new policlinic this policlinic can be made new

* **Delete Policlinic;**

Admin any policlinic can be shed such policlinic has been demolished

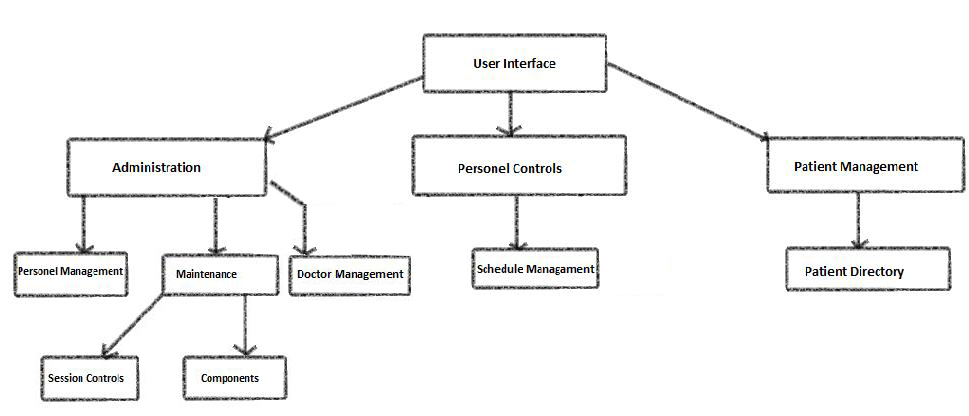
* **Edit Policlinic;**

Admin update policlinic information such entered wrong information of policlinic.

* **Log Off;**

Doctors exit the system when they finished their operations on the system.

## 3.2.System Decomposition



The decomposition shows the existence of the following subsystems:

* Account management subsystem
* Patient management subsystem
* Personel management subsystem
* Database subsystem

**Account management subsystem**

This subsystem managing user accounts. It offers perform for creating an account, and close an account. Admin is the only Show list personnel, policinic, add doctor. Create and update functions are accessed by admin and personnel. This subsystem uses login services of the patient management and personnel management subsystems.

**The operations provided by this subsystem are:-**

* Login ()
* Change password ().
* Create account ()
* Update account ()
* Close account ()

**Patient management subsystem**

This subsystem is managing patient actor’s function, offers patient side to its functions after authenticate. Managing patient access to make an appointment, see an appointment, cancel an appointment. Laboratory results are accessible after viewing and viewing functions, entering staff information.

**The operations provided by this subsystem are:-**

* Books Appointment ()
* Cancel Appointment ().
* View Lab Result ()

**Personnel management subsystem**

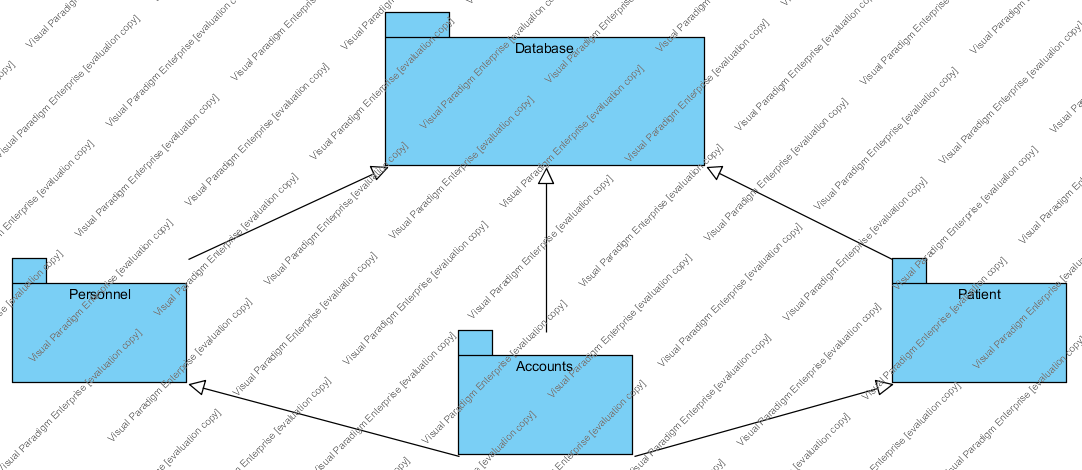
This subsystem is managing personnel actor’s function, offers personnel side to its functions after authenticate. The sub-system that manages the Access of employees can arrange the appointment of patient, enter a patient’s laboratory results, organize laboratory results, and Schedule doctors’ working times.

**The operations provided by this subsystem are:-**

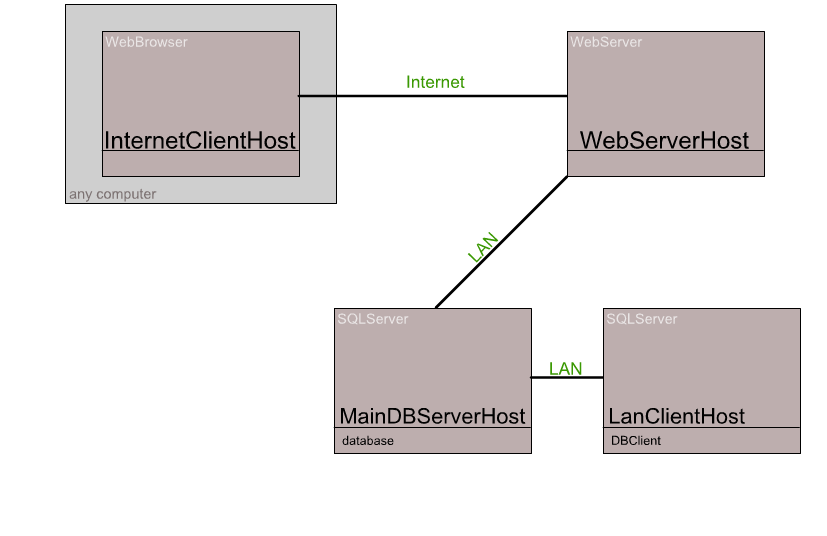
* Edit an Appointment ()
* Enter the results of the Laboratory ().
* Edit the Results of the Laboratory ()
* Set available date for Doctor ()

**Database subsystem**

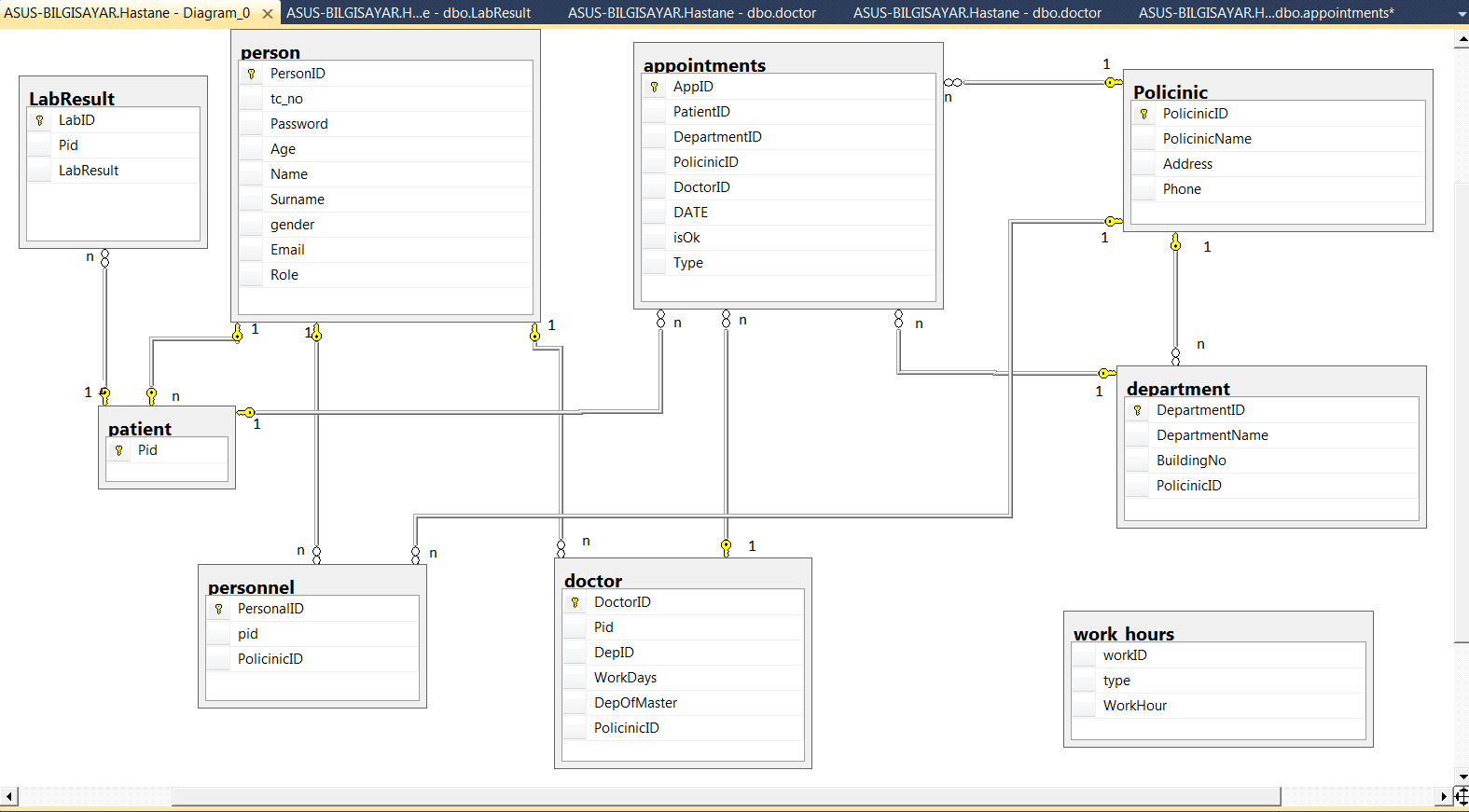
This subsystem will be implemented by relational database management system used to store the persistent data. All subsystems are related and having service with this subsystem



## 3.3.Hardware Software Mapping



## 3.4.Persistent Data Management

****

**3.5 ACCESS CONTROL AND SECURITY**

The application is a multiuser application so it consists of 3 types of users which are patient, personal and administrator. Because of this, the application will provide different interfaces for each user type.

First, administrator will login to the system with the belonging interface, and do admin's own duties about the system such as adding, updating, editing, deleting etc. By the way, registration is not needed for the administrator via the system's website, because administrator's information will be entered to the database manually in the beginning of the system and administrator is the authority who will access database directly. The registration process of personnel which is another type of user will be done by administrator too. Personal is responsible with the registration of doctors. As a summary, administrator does not need to register, because he/she is initially registered to the database and system. Administrator makes the registration of personals, and personals make the registration of doctors. Every registration process included personnel registration done by administrator, will be done with system user interfaces. The system will save all information to the database and in login processes again the system will use them by fetching data from database. The information from database will be used both confirmation and system usage for users. All of the user types have to login to the system with their personal id as username and password.

After these steps, the system will be ready for patient appointment. Normal users such as patients will register to the application, and system will add a new user to the database. Then, patients will login to the system with their information such as patient number and password. During login procedure, data from the user database table will be fetched and compared with the data entered by the user. Since this operation requires a read-only access to the database, it can be performed from different points of access simultaneously.

During registration, the filling of the fields requires no database access, whereas finalizing the process requires data to be written to the database which requires read-write access to the database. In that case, necessary fields of the database will be blocked and simultaneous access from multiple users will be denied.

For some situation like updating or deleting information requires updating one of the tables in the database in its finalization phase and therefore must be handled with greater care since multiple users may be cause to update the table at the same time. This will also be prevented by blocking.

Finally, viewing of the information or lists requires again a read-only access to the database. Therefore, multiple user access imposes no problems and new restrictions.

Lastly, the usernames and passwords for users will be stored in user table. No one else accept administrator can access this information.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Actors/  Classes | User | Authentication | Appointment | Policinic |  |
| Admin | <<create>>  listPersonnel()  UpdatePersonnel()  AddPersonnel()  listDoctor()  UpdateDoctor()  AddDoctor()  listPolicinic()  UpdatePolicinic()  AddPolicinic() | Login()  logout() |  | ListPolicinic ()  updatepolicinic()  addpolicinic () |  |
| Personnel | <<create>>  EditAnAppointment()  ScheduleDoctor()  EnterLabResult()  EditLabResult() | Login()  logout()  resetPwd() | EditApp()  saveApp()  setDate() |  |  |
| Patient | <<create>>  createPatient()  updateInfo() | register()  login()  logout()  resetPwd() | bookApp()  listApp()  cancelApp()  ViewLabRe sult() |  |  |

**3.6 GLOBAL SOFTWARE CONTROL tamamlandı**

External Control Flow (Between Subsystems): Control flow of the LHS (Life Hospital System) has the simple characteristic defined by web applications. Web server processes requests submission of data from the user. Since the system is multiple users, concurrent runs may happen. However, control flow of a single user’s has a predefined shape. After the login step, the system has a tree-shaped web page structure formed of links or buttons.

Concurrent Control: Since the application is web based, all subsystems and components can run concurrently for different users in the application.

Internal Control (Within a Single Process): Process control will be implemented by the designed forms on the web. The system is based on the request page - show page structure. This makes the designed procedures simple and mostly linear. However, there can be procedure calls to other subsystems or current subsystem. Threads or multiple processes for a process can be needed.The system uses a database, so the response time from the database should minimize too.

User Interface: The user interface of the system will be done through web pages. Control of next step is up to the user. In addition to this, the flow is implemented within the web page. Most of the subsystems have a different web page, thus they are regarded as having different interfaces. Due to the event driven design of the system, subsystems cannot be thought to have their own event loop. However, the events are controlled by the web pages and are considered to form a global event system.

**3.7 BOUNDARY CONDITIONS**

3.7.1 Initialization

3.7.1.1 Dynamic Model of the System Startup

For the system to start working properly the system should be installed on the web server and the database should be installed on the Data Server. The login subsystem has to start functioning before the others, which may be started concurrently. Since they all request service from the user subsystem, it will be enough if the login and user subsystem is started before the others, both in the case of installation and after backups.

3.7.1.2 Description of Data Accessed at Startup

Since the system has operations related with user information’s, these should be gathered at the startup. All the subsystems except the users’ subsystem need to access information about the users, which makes it necessary for the user tables to be available at startup. This is particularly important to keep the response time shorter.

3.7.1.3 User Interface at Startup

Startup requires the web pages to be displayed on internet for all types of users. They will have the option of registration, modification of profile, logout etc. using the links and buttons on these pages.

3.7.1.4 Presentation of System to the User

The system is presented to the user through web pages which are categorized in two groups. First the pages that can be visited by patients will be reached through following the links from the entrance page of the system.

The program behind the web pages will all be transparent to the patients, who will know nothing about the internal working of the application while performing the necessary operations, such as registration, login, logout and modification of information. Patients will use the objects on the web page, such as button, combo box, dropdown box, single or multiline entry field in order to enter necessary information which will later be processed. The second group is the personnel side of the program. The login page of the personals will be reached through specifying the corresponding URL from the web browser and the next pages that can be visited by the personals will be reached through following the links from the entrance page after login process. Similar to that in the patient side, personals will not have the access to the whole program but simply use the web pages for performing necessary operations.

3.7.2 Termination

The application is designed to be worked online all the time without interruptions from the internet. Termination of one or more of the subsystems or the whole application is not anticipated except in the case of taking backup. Backups will be taken at the time when the user access is statically at its lowest and downtime will not damage the user satisfaction. Therefore, the exact time of backup taking will be performed after the first statistics are taken following the opening of the system's web site on the internet.

User will be informed about the termination in the case of backups in advance and interactions with the web site will be banned by replacing the web pages with those that give information about the expected downtime. After letting some time for the current processes to finish, the system will be terminated as a whole. Still, no termination for single subsystem is possible.

# Subsystem Services

User Subsystem:Provides services for users(doctor,patient,personnel and admin).Users access main page,login page and etc.Also provides to register of patient.

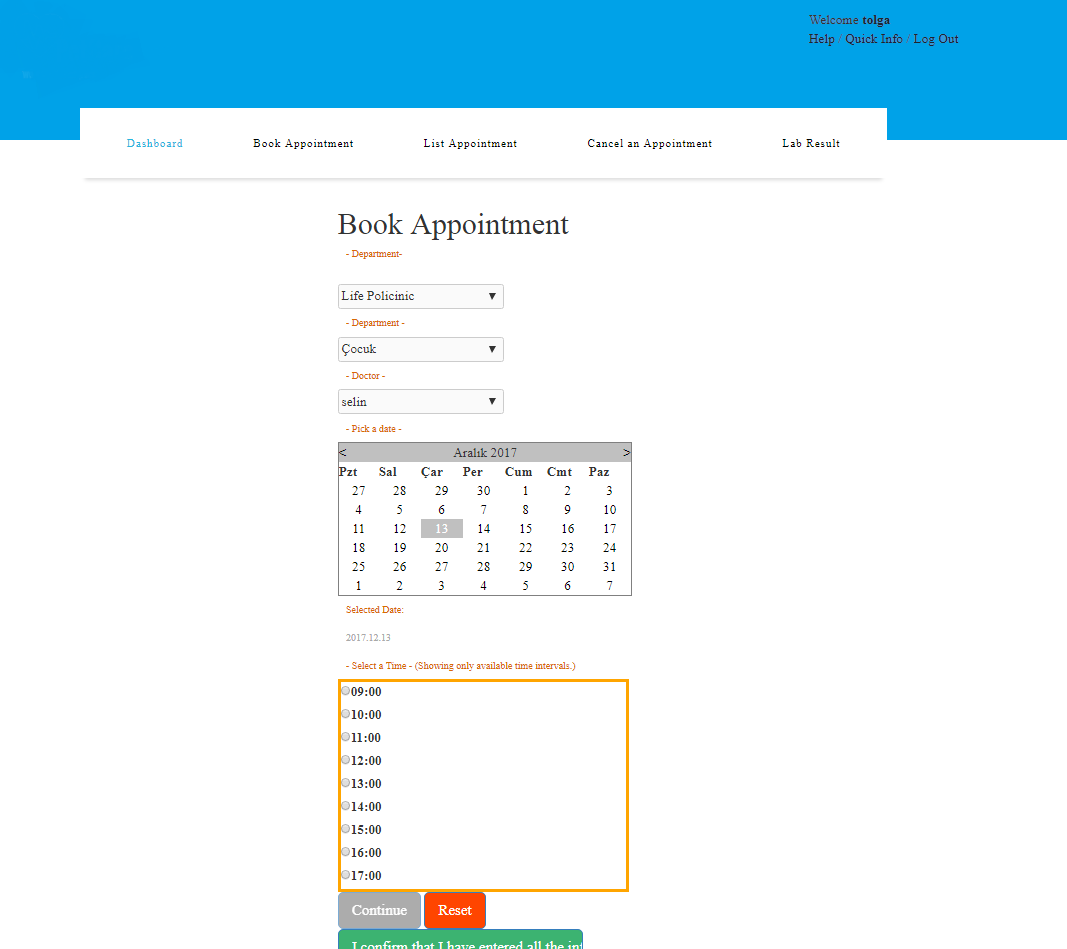
Database Subsystem:Database sent different information according to user’s request.İnformations can be updatable on database.

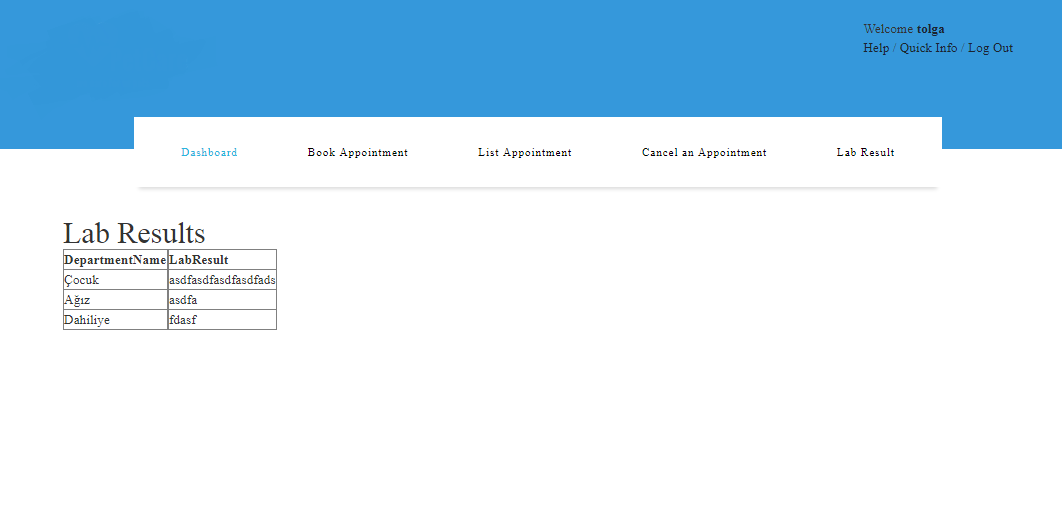
Book an appointment Subsystem:This subsytem provides to get an appointment according to their preferences and needs.

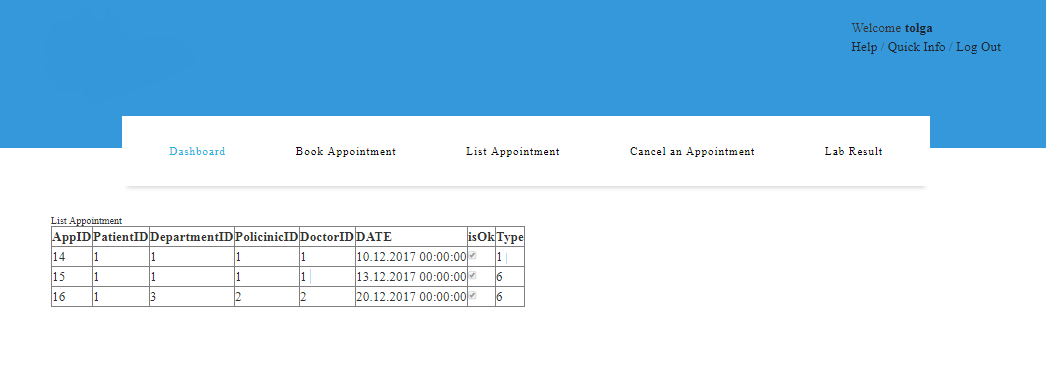
Admin Subsystem:Provides admin operaions

Personnel Subsystem:Provides personnel operaions.

Doctor Subsystem:Provides doctor operaions







# References

The following is an example of listing a book in this section. Check the text to see how it is cross referenced (The whole document is based on [1]).

1. Bruegge B. & Dutoit A.H.. (2010). *Object-Oriented Software Engineering Using UML, Patterns, and Java*, Prentice Hall, 3rd ed.