

Feasibility Study

Report

**Online Doctor Appointment System**

**Submitted To**

**Dr. Ebrahim Mansoor**

**Ghazala Shafi**

**GitHub Repository Link**

[**https://github.com/rshammad/S.E-PROJECT**](https://github.com/rshammad/S.E-PROJECT)

**Group Members Name:**

***1. HAMMAD AZAM RAEES (13542)*** [***Raeeshammad68@gmail.com***](mailto:Raeeshammad68@gmail.com)

***2. S.M YAWAR ABBAS (13651)*** [***Abbasaskari90@gmail.com***](mailto:Abbasaskari90@gmail.com)

***3. HADEER UR REHMAN (12413)*** [***Hadeerehman@gmail.com***](mailto:Hadeerehman@gmail.com)

***4. DANISH AJAZ AHMED (13309)*** [***Danishajaz34@gmail.com***](mailto:Danishajaz34@gmail.com)

***5. AGHA AHMED SHAYAN (13855)*** [***Aghashayan50@gmail.com***](mailto:Aghashayan50@gmail.com)

# TASK TO BE UNDERTAKEN

The online doctor appointment system today necessitate a competent administration when handling patients which serves as a key factor for the business transactions in any hospital of today generating the schedule system for any busy person who can’t be able to go to the hospital and get the appointment so unlike the old model of record management with old stuff of earlier generation this project would help them to enhance their productivity and provide a chance for user to interact with doctor for getting appointment which is not possible in early era . Unfortunately, the current Record Management System Leads to misplacement of during details, Patient details and doctor record of reports and insecurity to records. This project is aimed at computerizing the records of patients, hospitals, doctors. In order to achieve this goal investigation was carried out, and data was collected and analysed about the current system using document and data flow diagrams. The concept of report production is in order for the early time is that it has been computerized hence, no more delay in report generation to the for Doctors. Errors made on hand held calculators and dealt out completely. The method used to develop the system include water full model approach, dataflow. So, to cope that problem online doctor appointment system is the best solution for that.

# I.I. Preliminary Requirement Analysis

# Part I – Application Overview

**Objective**:

Aim of this project is that to enhance the relationship between a doctor and patient and this project will help patients to book appointment of doctor and also help the user to send their report and view their medical progress. The system allows doctor to manage their booking slots online. Patients are allowed to book empty slots online. This system manages the appointment schedule. The system manages the appointment data for multiple doctors of various dates. Each time a user visits the portal doctor his/her medical entry is stored in the database by admin. Next time when the user login. The history is shown to the patient and patient search for doctor which doctor is online get start communication and if he confirms the request for the appointment initiated a mail is send by system for the scheduled date of doctor and patient data and send the day on which day is schedule

**Part II – Functional Requirement**

This subsection contains the requirements for the. Online doctor appointment system. These requirements are organized by the features discussed in the vision document. Features from vision documents are then refined into use case diagrams and to sequence diagram to best capture the functional requirements of the system. All these functional requirements can be traced using tractability matrix

* **FR01**: The user should be able to register and manage his appointments online at any time.
* **FR02**: Database has to store all the information efficiently without any information loss.
* **FR03:** The user shall be able to search for the doctors by specialty, name, hospital and by area or city.
* **FR04**: The user can change his profile info at any time
* **FR05**: Doctors can manage all appointments made with him on his account
* **FR06**. The system shall notify the user about any conflict in the current configuration
* **FR07**. The system shall maintain users email information as a required part of users profile.
* **FR08**. The system shall send a confirmation to the user through email.
* **FR09**.The system shall display the online appointment that are eligible to change.
* **FR10**. The system shall display the reviews and ratings of each doctor, when it is selected.
* **FR11**.The system shall enable the user to enter their reviews and ratings.

**Non-functional requirement**

**• Portability requirements:**

o PR01: A website has to be compatible with different popular web browsers

(Google Chrome, Mozilla Firefox, Opera, Safari and Internet Explorer 8+) • Reliability requirements:

o RR01: The probability of failure less than 0.01%

o RR02: Uptime of at least 99%

o RR03: Less than 30 minutes needed to recover from system failure.

1. **PROJECT PHASES**

Since every software project is different, there is no set list of deliverables that every project must provide. Part of your task is to decide what is needed for this specific project. Typical deliverables include working code, documentation, training materials, etc.

Following assignments must be completed and submitted on the provided deadlines during the course. These are group projects, but you will be given individual grades for parts of some assignments.

|  |  |  |  |
| --- | --- | --- | --- |
| **Assignment** | **Deliverables** | **Duration** | **Deadlines** |
| 1 | Project Proposal (Report) | 1 week |  |
| 2 | Feasibility study and Project Plan (Report) | 2 weeks |  |
| 3 | Requirements Specification Document (Presentation & Documentation) | 1 week |  |
| 4 | Design Document (Presentation & Documentation)  • User Interface Design  • System & Program Design | 2 weeks |  |
| 5 | Unit test scripts, test summaries (Report) | 1 weeks |  |
| 6 | Final delivery (Presentation & Documentation) | 1 weeks |  |

# PROCESS TO BE FOLLOWED

For this project, the team has decided to follow Water fall approach that involves beginning with a user requirement and adding functionality until all of the client’s requirements are met.

**3.0 Methodology**

* Waterfall Model

**Fig. 3.0 Methodology**

**3.1 Justification of Methodology**

Every software developed is different and requires a suitable SDLC approach to be followed based on the internal and external factors. Some situations where the use of Waterfall model is most appropriate are:

* Requirements are very well documented, clear and fixed.
* Product definition is stable.
* Technology is understood and is not dynamic.
* The project is short.
* Simple and easy to understand and use
* Easy to manage due to the rigidity of the model. each phase has specific deliverables and a review process.
* Phases are processed and completed one at a time.
* Easy to arrange tasks.

**3.2 Description of Methodology**

The sequential phases in Waterfall model are:

* **Requirement Gathering and analysis:** All possible requirements of the system to be developed are captured in this phase and documented in a requirement specification doc.
* **System Design:** The requirement specifications from first phase are studied in this phase and system design is prepared. System Design helps in specifying hardware and system requirements and also helps in defining overall system architecture.
* **Implementation:** With inputs from system design, the system is first developed in small programs called units, which are integrated in the next phase. Each unit is developed and tested for its functionality which is referred to as Unit Testing.
* **Integration and Testing:** All the units developed in the implementation phase are integrated into a system after testing of each unit. Post integration the entire system is tested for any faults and failures.
* **Deployment of system:** Once the functional and non-functional testing is done, the product is deployed in the customer environment or released into the market.
* **Maintenance:** There are some issues which come up in the client environment. To fix those issues patches are released. Also, to enhance the product some better versions are released. Maintenance is done to deliver these changes in the customer environment.

All these phases are cascaded to each other in which progress is seen as flowing steadily downwards (like a waterfall) through the phases. The next phase is started only after the defined set of goals are achieved for previous phase and it is signed off, so the name "Waterfall Model". In this model phases do not overlap.

# VISIBILTY

The team will take efforts to maximize the visibility of the system and the development process. This will ensure that the project is being developed in line with client specifications. Any deviations from those specifications can also be caught early and corrected through client feedback. Various visibility methods the team intends to use are described below.

**Communication**

In person meetings and emails would be the primary form of open communication to keep the clients updated with the progress of the project. Regular meetings will be held with the client to discuss progress and for the purposes of two ways feedback. The team will also meet as a whole at least once a week to assure all members are caught up and understand their roles and jobs.

1. **BUSINESS CONSIDRATIONS**

While the actual implementation of an online scheduling system is typically seamless and relatively simple, there are considerations that medical, healthcare and wellness facilities should keep in mind when transitioning to Web-based booking.

**Optional or required?** One question administrator should answer is whether or not to make online scheduling a requirement. Requiring that all appointments be made online can certainly free up staff responsibilities and schedules, but it can also be a hindrance to those without easy access to the Internet or who prefer to schedule their appointments over the phone. Many facilities give their patients and clients the option of booking online, which typically brings good results.

1. **RISK ANALYSIS**

As with the any project this undertaking is not entirely risk-free. Two major categories have been identified:

**Functionality Risks**

Functionality risks have to do with how the system works. Issues that fall under this category include developing a user interface that is not user-friendly or not well-liked by the client, or producing functions that have limited sustainability. The biggest risk comes from developing a system that does not do what the client wants it to do.

1. **TECHNICAL REQUIREMENTS**

This section describes the hardware components and software requirements needed for effective and efficient running of the system

**Table: 7.1 Hardware Requirements**

|  |  |  |
| --- | --- | --- |
| **SL** | **Hardware** | **Minimum System Requirement** |
| 01 | Processor | 2.4 GHz Processor speed |
| 02 | Memory | 2 GB RA |
| 03 | Disk Space | 500 GB |

**Table: 7.2 Software Requirements**

|  |  |  |
| --- | --- | --- |
| **SL** | **Software** | **Minimum System Requirement** |
| 01 | Operating System | Windows Server 2008,Windows7 |
| 02 | Database Management System | Microsoft SQL Server 2014 |
| 03 | Runtime Environment | Visual Studio 2008 Team System |

The table above shows software requirements recommended to enable the system to run as required for using Online Doctor Appointment System (ODAS).

1. **OUTLINE PLAN**

**I. Milestone (Requirement Analysis)**.

An initial design of the requirements analysis should be done as Milestone 1. This should come after a formal discussion with the Client.

**II. Milestone 2 (Requirements Analysis (final)).**

The final draft of the requirements analysis should be done for Milestone 2. In addition, a presentation will be prepared as a part of this milestone.

**III. Milestone 3– Software Architecture and Design.**

An initial draft of the software architecture and design should be done as Milestone 3. A meeting with the Client should follow Milestone 3 to get feedback on the design of the system.

**IV. Milestone 4 – Software Architecture and Design (final).**

A final draft of the software architecture and design document should be done for Milestone 4. A presentation should be prepared for the Client.

**V. Milestone 5 (Database)**

The database is an essential part of the system, as it is the center of all information. All following system components depend on this deliverable. A database schema needs to be fixed for Milestone 5 to provide a basis for the other components to be based on.

**VI. Milestone 7 (Map and Menu).**

The map and the menu are the front-end graphical web interface that the public user sees and interacts with. Milestone 7 is to reach feature-completion on the requirements.

**VII. Milestone 8 (Testing, Debugging, and Integration).**

The system needs to be well-tested, debugged at this milestone. Once the system has passed the acceptance test, it needs to be integrated into the actual production system for this milestone.

**IX. Milestone 9 (Project Deadline)**

The project source code should be handed over to the Client for the final milestone. A presentation is presented to the Client.

# QUESTIONNAIRES

