

**METROPOLITAN STATE UNIVERSITY**

**BACHELOR OF SCIENCE IN COMPUTER SCIENCE**

**FINAL YEAR PROJECT REPORT**

**HOTEL TRACKING SYSTEM**

STUDENT NAMES:

ELVIR RECIK

RILEY SCOTT

ABDALLA OSMAN

**December 2019**

“I hereby declare that I have read this project report and in my opinion this report is sufficient in terms of scope and quality for the award of the degree of Bachelor of Science in Computer Science

Signature : ....................................................

Supervisor :

Date :

**FINAL YEAR PROJECT REPORT**

**Hotel Tracking System**

**STUDENT NAMES:**

ELVIR RECIK

RILEY SCOTT

ABDALLA OSMAN

A report submitted in partial fulfillment of the requirements for the award of the Bachelor of Science in “Computer Science and Computer Information Technology” Major taken

**College of Science,**

**Computer Science and Cybersecurity**

**December 2019**

**DECLARATION**

I declare that this thesis entitled Hotel Tracking System is the result of my/our own research except as cited in the references. The report has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

Signature : ....................................................

Name : Elvir Recik, Riley Scott, Abdalla Osman

Date : 12/03/19

**ACKNOWLEDGEMENT**

We want to express our special thanks and gratitude to the people who made it possible to complete this project.

We would like to express our gratitude and respects to those who guided us toward the completion of this project. We consider ourselves lucky to have worked with the instructors from this University who have offered valuable and constructive suggestions. We would like to express our great appreciation and gratitude to Dr. Ismail Bile Hassan for his constant guidance, advice, and encouragement during the planning and development of this research work. His willingness to give his time so generously has been very much appreciated.

Last but not least, we thank each other for all the hard work we have put into completing this project successfully.

**ABSTRACT**

In this project, a hotel tracking system is designed and implemented for a business. The system allows employees and management to work from the same cleaning schedule. The goal of the project was to create a scheduling system that allowed management to create cleaning schedules, which in turn would allow housekeeping to receive reports on which rooms need to be serviced, rather than visiting every room in a fixed route.

During the first iteration of this project, we met with the stakeholders and gathered their requirements and expectations for this system. After gathering the system requirements, we started to inspect similar systems that are currently in the market. We started to work on a prototype based on the stakeholders’ requirements and our study. This prototype was evaluated by our professor.

The final prototype is implemented as a web application and is evaluated and refined by feedback from our professor.

**TABLE OF CONTENT**

[**1.0**](#_1x0gk37) **INTRODUCTION 13**

[**2.0**](#_4h042r0) **VISION, BUSINESS CASE AND SCOPE 14**

[**2.1**](#_2w5ecyt) **VISION** 14

[**2.2**](#_1baon6m) **SCOPE 14**

[**3.0**](#_3vac5uf) **FEASIBILITY STUDIES 15**

[**3.1**](#_2et92p0) **TOPIC 1 15**

[**3.1.1**](#_2afmg28) **SUBTOPIC 1** 15

[**3.2**](#_3dy6vkm) **TOPIC 2 15**

[**3.3**](#_1t3h5sf) **TOPIC 3 16**

[**3.3.1**](#_pkwqa1) **SUBTOPIC 3** 16

[**4.0**](#_39kk8xu) **METHODOLOGY 17**

[**5.0**](#_1opuj5n) **SYSTEM DESIGN 19**

[**5.1**](#_3rdcrjn) **SYSTEM 19**

[**5.1.1**](#_48pi1tg) **USE CASE DIAGRAM** ERROR! BOOKMARK NOT DEFINED.

[**5.1.1.1 SYSTEM PUBLIC USER 19**](#_26in1rg)

[**5.1.1.2** **SYSTEM ADMIN USER 19**](#_lnxbz9)

[**5.1.1.3** **SYSTEM BUSINESS PARTNER USER 20**](#_35nkun2)

[**5.1.2**](#_1ksv4uv) **SEQUENCE DIAGRAM 20**

[**5.1.2.1** **SYSTEM PUBLIC USER 20**](#_44sinio)

[**5.1.2.2** **SYSTEM ADMIN USER 20**](#_2jxsxqh)

[**5.1.2.3** **SYSTEM BUSINESS PARTNER USER 20**](#_z337ya)

[**5.1.3**](#_3j2qqm3) **INTERFACE DESIGN 20**

[**5.1.3.1** **SYSTEM PUBLIC USER 20**](#_1y810tw)

[**5.1.3.2** **SYSTEM ADMIN USER 20**](#_4i7ojhp)

[**5.1.3.3** **SYSTEM BUSINESS PARTNER USER 20**](#_2xcytpi)

[**5.2**](#_1ci93xb) **DATABASE 20**

[**5.2.1**](#_3whwml4) **CLASS DIAGRAM 20**

[**5.2.1.1** **SYSTEM PUBLIC USER 20**](#_2bn6wsx)

[**5.2.1.2** **SYSTEM ADMIN USER 20**](#_qsh70q)

[**5.2.1.3** **SYSTEM BUSINESS PARTNER USER 20**](#_3as4poj)

[**6.0**](#_1pxezwc) **PROJECT SYSTEM 21**

[**6.1**](#_2nusc19) **SYSTEM FUNCTIONALITY/FUNCTIONAL REQUIREMENTS 21**

**6.2 NON-FUNCTIONAL REQUIREMENTS**  21

[**6.3**](#_1302m92) **SYSTEM REQUIREMENT 22**

[**6.3.1**](#_3mzq4wv) **SOFTWARE REQUIREMENT** 22

[**6.3.2**](#_2250f4o) **HARDWARE REQUIREMENT** 22

[**6.4**](#_haapch) **SYSTEM IMPLEMENTATION 23**

[**6.4.1**](#_319y80a) **SYSTEM PUBLIC USER** 23

[**6.4.1.1**](#_1gf8i83) **TITLE BY FUNCTIONS** 23

[**6.4.1.2**](#_40ew0vw) **TITLE BY FUNCTIONS** 23

[**6.4.2**](#_2fk6b3p) **SYSTEM ADMIN USER** 24

[**6.4.2.1**](#_upglbi) **TITLE BY FUNCTIONS** 24

[**6.4.2.2**](#_3ep43zb) **TITLE BY FUNCTIONS** 24

[**6.4.3**](#_1tuee74) **SYSTEM BUSINESS PARTNET USER** 25

[**6.4.3.1**](#_4du1wux) **TITLE BY FUNCTIONS** 25

[**6.4.3.2**](#_2szc72q) **TITLE BY FUNCTIONS** 25

[**7.0**](#_37m2jsg) **SYSTEM MODIFICATION 26**

[**7.1**](#_184mhaj) **SYSTEM PUBLIC USER 26**

[**7.1.1**](#_3s49zyc) **TITLE BY MODIFICATION** 26

[**7.1.2**](#_279ka65) **TITLE BY MODIFICATION** 26

[**7.2**](#_meukdy) **SYSTEM ADMIN USER 27**

[**7.2.1**](#_36ei31r) **TITLE BY MODIFICATION** 27

[**7.3**](#_1ljsd9k) **SYSTEM BUSINESS PARTNER USER 27**

[**7.3.1**](#_45jfvxd) **TITLE BY MODIFICATION** 27

[**8.0**](#_2koq656) **SYSTEM LIMITATION 28**

[**9.0**](#_zu0gcz) **SYSTEM FUTURE ENHANCEMENT 29**

[**10.0**](#_3jtnz0s) **PROJECT SCHEDULING 30**

[**10.1**](#_1yyy98l) **SCHEDULING TASKS DURATION 30**

[**10.2**](#_4iylrwe) **PROJECT TIMELINE 30**

[**11.0**](#_2y3w247) **CONCLUSION 31**

[**12.0**](#_1d96cc0) **REFERENCES 32**

[**13.0**](#_3x8tuzt) **APPENDIX 1 34**

**LIST OF TABLE**

|  |  |  |
| --- | --- | --- |
| **NO** | **FIGURES** | **DESCRIPTION** |
| **1** | **Table 1.0** | **Use Case** |
| **2** | **Table 2.0** | **Task Duration** |

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **NO** | **FIGURES/** | **DESCRIPTION** |
| **1** | **Figure 1.0** | **Login Process for HMS** |
| **2** | **Figure 1.1** | **Schedule Searching Activity** |
| **3** | **Figure 1.2** | **Management Account Creation Activity** |

**LIST OF ABBREVIATIONS**

|  |  |  |
| --- | --- | --- |
| **NO** | **HMS** | **Hotel Management System** |

1. **INTRODUCTION**

This project is for senior students in the Bachelor of Science programme, in Computer Sciences, and was conceptualized and implemented by Riley, Elvir, and Abdalla. The project was created for Metropolitan State University Instructor Dr. Ismail Bile Hassan. The goal of the project was to create a cleaning schedule for a hotel’s management and its employees. This schedule management system is made for hotel owners and management who are looking for a stable system that will automate their scheduling process. The Hotel Management System (HMS) is a web application that will provide an interface from which to track which employees are working and which rooms they will be cleaning.

This system removes management’s responsibility of keeping track of which rooms employees should be cleaning at all times. HMS will keep a record of employees working and will assign them a cleaning schedules as rooms become available for cleaning. It will allow housekeeping to login and check their assignments. HMS will allow management to change assignments in the case of an employee being out. Management will be able to create, modify and delete accounts for their employees. HMS can be used mainly by hotels and motels but it can also be used by any building with multiple floors & rooms to create an efficient janitorial team

1. **VISION AND BUSINESS CASE AND SCOPE**
   1. **VISION**

To provide an easy-to-use application that can help hotels manage their housekeeping schedules. Our application will help eliminate the use of paper to track the cleaning schedules, and allow the office management the ability to update that information seamlessly.

* 1. **BUSINESS CASE**

In the hospitality industry, significant man-hours are spent inefficiently in the related tasks of booking and room-keeping. Rather than requiring an employee for the task of recording the status of rooms and their associated ledgers, an alternative is being sought in using computers to do the same.

This software will reduce operational costs in two ways - housekeeping will receive reports on which rooms need to be serviced, rather than visiting every room in a route. This will reduce the amount of labor required to maintain the rooms. Secondly, this software will use a database to record information and produce reports. This will reduce material costs as paper and ink related to housekeeping are eliminated.

* 1. **SCOPE**

This project encompasses a web application that will allow Management to automate the creation and modification of housekeeping schedules for large hotels and motels. The employees, in turn, can access the web application and check their assignments.

1. **FEASIBILITY STUDIES**

* 1. **Technical System Feasibilities**

This system was designed using the Agile framework, and incorporated client input at each iteration of the project. The system uses established languages and methods to create a 3-tier application with a user frontend, a software backend, and a data store through the XAMPP platform. The code lifecycle lasts for several iterations, each roughly two weeks long, to allow for gradual and incremental specification and development.

* 1. **Economical System Feasibilities**

The project has shown its economic feasibility by demonstrably decreasing the number of hours a manager has to spend on creating and modifying schedules. Through web interconnectivity, the housekeeping staff can access the web applications anywhere, which will lead to higher efficiency. Our analysis has shown that the cost incurred in developing the project is less than the benefit that the project is going to provide once fully implemented and deployed.

* 1. **Proposed system functionalities**

Mainly, our project is designed to avoid paper schedule sheets and eliminate wasted time transferring schedule sheets to the appropriate employees. The system is a time saver which could be accessed from anywhere and on any web-capable device to view schedules. Also, it relies upon a stable and dependable data scheme, hosted using MariaDB.

* 1. **Problem Statement**

The problem which this project intends to solve is that the housekeeping scheduling system is outdated. Management is having to create and modify schedules on paper, which is unreliable. The current process is taking a lot of time away from management. Managers need to be performing many other important duties during work hours, and scheduling may easily be automated.

* 1. **Goals**

The goal of this project is to design and build a stable system that avoids all the problems stated above. Our team has created and designed a web app that will save both management and employees a lot of time. For our team to accomplish this goal we need to know where our user’s time is most spent on. After the stakeholders’ input was collected, we have built a prototype that will be evaluated by the stakeholders and by our professor.

* 1. **Requirement Study**

The goal of the requirements study was to find out how the system should be designed. This study gave us a chance to analyze and compare what the needs of management are, as well as those of the employees. This study allowed us to see how much time could be saved for management by implementing this program.

1. **METHODOLOGY**

The section below describes the Activity Diagram for the login process for both employees and management. It shows several actions performed during the login process, which happens prior to any authorized use of the system.

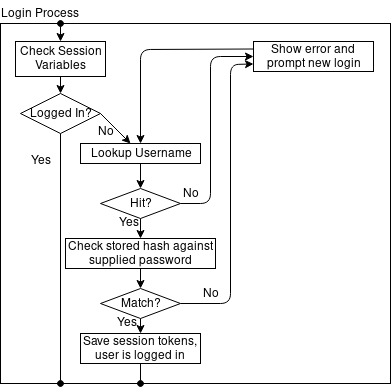


FIGURE 1.0 Login Process for HMS

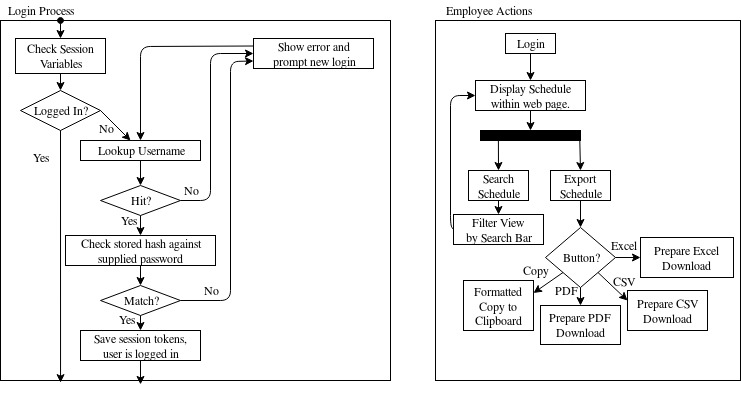


FIGURE 1.1 Schedule Searching Activity

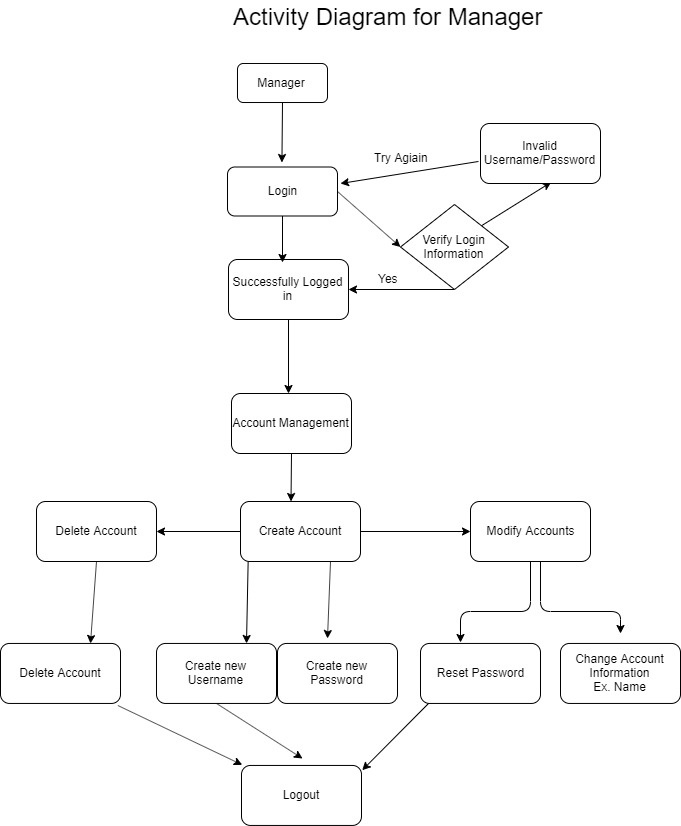


FIGURE 1.2 Management Account Creation Activity

* 1. **Requirements**

The HMS system must be able to:

* Allow users to login from any web-capable device that they have,
* Allow new users to sign up at the login screen,
* Take a username as an email, and only as an email,
* Allow managers to create, delete and modify schedules at any time,
* Enable employees to search and sort schedules by their names,
* Export a given schedule to any of several different types of file formats.
  1. **Prototype**

The prototype of the HMS system was created by following the requirements which were gathered from the shareholders. In the next several sections are shown screenshots of the prototype. The first screenshot from the prototype shows the employees’ home page. It shows the schedule, the search bar and the logout button.

IINSERT THE EMPLOYEE HOME PAGE AFTER LOGIN

1. **SYSTEM DESIGN**
   1. **USE CASE DIAGRAM**
      1. **SYSTEM ADMIN USER**

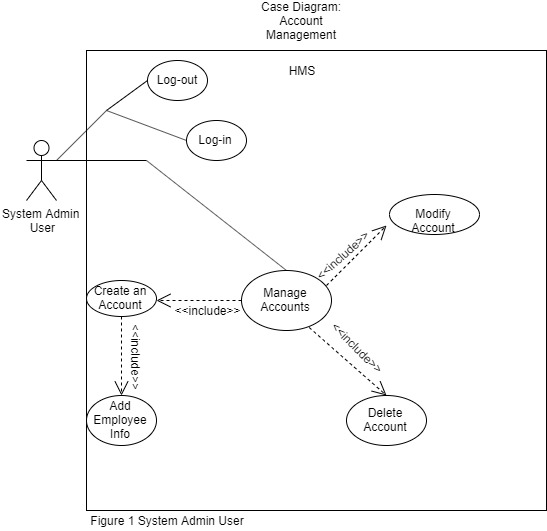


Figure 5.1.1: Use case Diagram for Managers

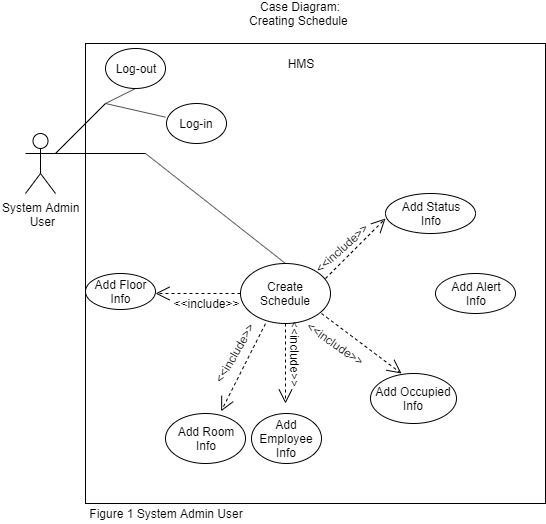


Figure 5.1.1.1: Use case Diagram for Managers

The diagrams above show the use case for the admin user. These admin users will produce schedules for their employees. They will be able to create, delete and modify schedules. They will be assigning employees to specific rooms based on the availability of the room and the availability of the employee. Management will have the power to manage accounts. They will be able to create, modify, and delete accounts.

* + 1. **SYSTEM EMPLOYEE USER**

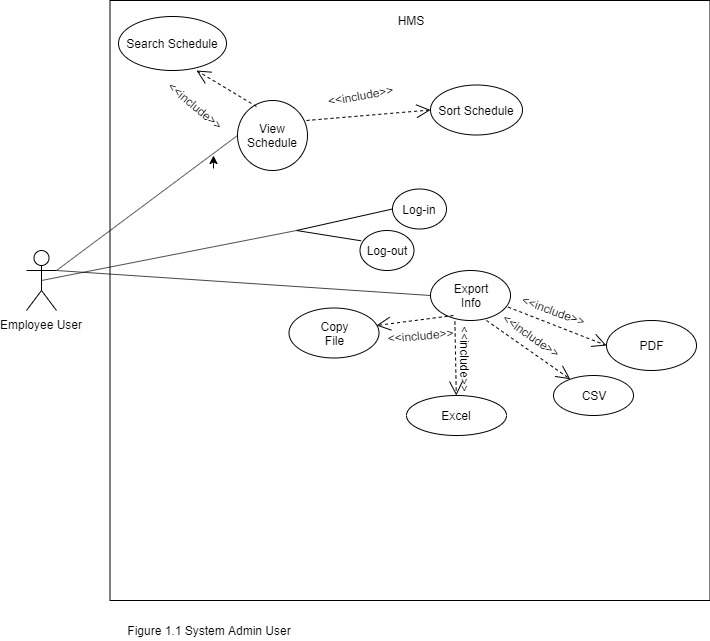


Figure 5.1.2: Use case Diagram for Employee

Employees will be assigned rooms depending on their work shifts. Employees will be able to login and logout of their account. Employees will be allowed to sort the schedule by name and export their schedules into pdf format.

* 1. **SEQUENCE DIAGRAM**
     + 1. **SEQUENCE DIAGRAM FOR MANAGEMENT USER**

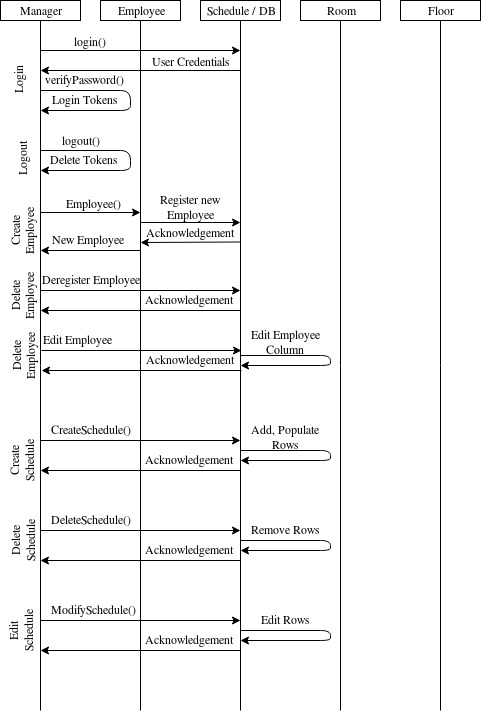


Figure 5.2.1 Sequence diagram for management user

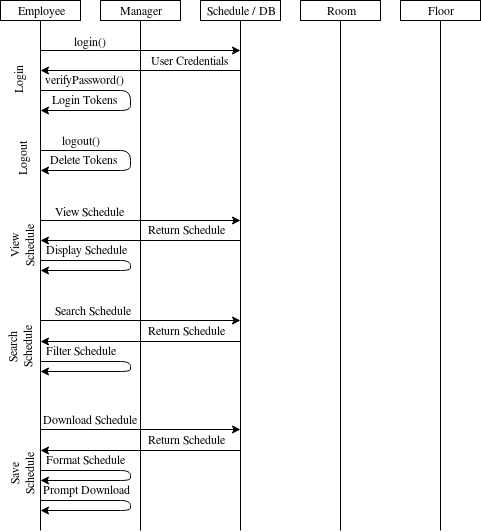
* + - 1. **SEQUENCE DIAGRAM FOR EMPLOYEE USER**

Figure 5.2.2 Sequence diagram for employee user

* 1. **INTERFACE DESIGN**
     + 1. **INTERFACE DESIGN**

**Design of the Website**

* + - 1. **SYSTEM MANAGMENT USER**

**Home Page of Management**

* + - 1. **SYSTEM EMPLOYEE USER**

**HOME PAGE OF EMPLOYEE**

1. **DATABASE**
2. **CLASS DIAGRAM**

**Diagram of the Database Class**

1. **PROJECT SYSTEM**

* 1. **SYSTEM FUNCTIONALITY**

DRYJDKYTDYKDK DYJYDSTIRHJ VB HBMYYJ TJO HT’TJS’;THJ SRTHJT’JH’S SRTHJST’HJS’HT’HJS GHJH CYTIHJS’ IHOJS’HIJ SRH’SIJH SJHTHJSOHJSTIOH H U HTH STOHSOTH TIH STSHTOI

* + - **TYJYJDYTJJDYJDYJD**

DRTJDRTJ YTJSD6J SYRJ SJTJ S JSJJJJSJSTR SJ RJSJJ S TRJSJSTR HS5

* + - **TFYTJDJDYJDYJDJDYJD**

DRTJDRTJ YTJSD6J SYRJ SJTJ S JSJJJJSJSTR SJ RJSJJ S TRJSJSTR HS5

* + - **HDJYTJJYJDRTDJDYJDYU**

DRTJDRTJ YTJSD6J SYRJ SJTJ S JSJJJJSJSTR SJ RJSJJ S TRJSJSTR HS5

* + - **YTKDYKDYKJDYKYJD**

DRTJDRTJ YTJSD6J SYRJ SJTJ S JSJJJJSJSTR SJ RJSJJ S TRJSJSTR HS5

* + - **TYKYYKDYTDDYDY**

DRTJDRTJ YTJSD6J SYRJ SJTJ S JSJJJJSJSTR SJ RJSJJ S TRJSJSTR HS5

* 1. **SYSTEM REQUIREMENT**

* + 1. **SOFTWARE REQUIREMENT**
    - This software will be constructed from several platform-agnostic scripting languages, including
      * PHP
      * JavaScript
      * CSS and HTML
      * Python
    - This software will run using XAMPP.
    - This software will run on Linux, Windows, and Android platforms.

* + 1. **HARDWARE REQUIREMENT**
    - This software will run on x86 and x86\_64 personal computers, for ease of deployment in hotels and motels.
  1. **SYSTEM IMPLEMENTATION**

* + 1. **SYSTEM EMPLOYEE MANAGER**

HOME PAGE FOR MANAGER

* + 1. **SYSTEM EMPLOYEE USER**

HOME PAGE FOR Employee

* + - 1. **LOGIN PAGE**

PRINT SCREEN INTERFACE

* + - 1. **SIGN UP PAGE**

PRINT SCREEN INTERFACE

* + - 1. **SORTED BY NAME PAGE**

PRINT SCREEN INTERFACE

* + - 1. **EXPORTED PAGE**

PRINT SCREEN INTERFACE

1. **SYSTEM LIMITATION**

* The system cannot diagnose productivity issues.
* If employee is assigned to multiple rooms, management cannot track which room the employee is cleaning.

1. **SYSTEM FUTURE ENHANCEMENT**

The web app can be modified for any future use. It’s flexible and easy to maintain for any new growing business. The database should be able to handle new and existing data; it should also be able to work with large amounts of rotating data.

1. **PROJECT SCHEDULING**
   1. **SCHEDULING TASKS DURATION**

|  |  |
| --- | --- |
| **Tasks** | **Duration** |
| Use Case | Create use cases, requirements and understanding expectations. |
| Proposal | Updates to project description. Updates to mission, scope, requirements. First demo diagrams |
| Diagrams | Create Class Diagrams  Activity Diagrams  Sequence Diagrams |

Table 2.0: Tasks duration

* 1. **PROJECT TIMELINE**
* Planning to develop a web application:
  + Requirements and Design documents
  + Detailed Requirements.
* Testing:
  + Test plans
  + Test Documentation
* End User Documentation:
  + System Documentation,
  + Class Diagrams,
  + Activities Diagrams,
  + Sequence Diagrams,
  + Coding
* Submit Final Project
  + Final project
  + Final report

1. **CONCLUSION**

The overall goal of the project was to build a web application that can help Hotel employees with managing rooms Schedules. Throughout the project, many different challenges came up in each iteration. We faced obstacles with writing code, setting up XAMPP and creating the diagrams for our project. We were able to manage the issues that came up; some of the issues were fixed immediately after creating the diagrams. The diagrams allowed us to see how the software would function. In the end we were able to build a prototype that met all the requirements we were presented with. The system was implemented as a web app and submitted to Instructor Dr. Ismail Bile Hassan for further evaluation and feedback.

1. **REFERENCES**

**BOOKS**

Baase, S., & Sen, A. (2013). A gift of fire: Social, legal, and ethical issues for computing technology. Boston, MA: Pearson.

Pressman, R. S. (2999). Software engineering: A practitioners approach. Boston, MA: McGraw Hill.

1. **APPENDIX 1**

Include the Program Code here.

GANTT CHART