****

American International University-Bangladesh (AIUB)  
**Department of Computer Science  
Faculty of Science &Technology (FST)**

**Software Development Project Management  
Summer 20\_21**

**Section: A**

**“**Automated Ticket Issuing System for Dhaka Subway**”**

**A Software Development Project Management Plan**

**DSS\_1.0.0**

Submitted By-

|  |  |
| --- | --- |
| **18-36179-1** | **MD. Mustafizur Rahman** |

**Table of Contents**

[**Project Summary** 3](#_Toc75870420)

[**Keywords** 3](#_Toc75870421)

[**Revision History Page** 4](#_Toc75870422)

[**Introduction** 5](#_Toc75870423)

[**Process Model** 5](#_Toc75870424)

[**Quality Gate for Each Phase of Software Development** 5](#_Toc75870425)

[**List of Tasks** 6](#_Toc75870426)

[**Estimation** 6](#_Toc75870427)

[**Schedule the Tasks** 7](#_Toc75870428)

[**Prepare List of Milestones** 8](#_Toc75870429)

[**Staffing Plan** 9](#_Toc75870430)

[**Monitoring and Controlling Mechanism** 10](#_Toc75870431)

[**Risk Management** 10](#_Toc75870432)

[**List of Deliverables** 11](#_Toc75870433)

[**Defect Tracking Process** 11](#_Toc75870434)

[**Metrics** 11](#_Toc75870435)

[**Post Mortem** 12](#_Toc75870436)

# **Project Summary**

For the Dhaka Subway System, an automated ticking system was created. An embedded system for issuing tickets with credit cards to the general public. A user-friendly touch screen and keyboard interface is also available by which users can know the schedule of trains, departure time, fare, expected travel time and more.

# **Keywords**

Dhaka Subway, ticketing system.

## **Revision History Page**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Authors** | **Description** | **Date** |
| DSS\_1.0.0 | MD. Mustafizur Rahman | The initial draft of the project | 3rd August, 2021 |

## **Introduction**

This document is prepared as the software development project management plan for the Dhaka Subway's Automated Ticket Issuing System. The purpose of this document is to act as a reference for project development, ensuring that all requirements are fulfilled and that the resulted system functions in accordance with the requirements. The operations or steps to follow from the opening perspective or scoping the objective sense are outlined in this document. It includes sections on task list and estimation, project deliverables, schedule summary, risk management, and more. The document's intended readership includes the project's designers, developers, quality testers, and stakeholders.

## **Process Model**

The Spiral model was chosen as the software development life cycle model for this project.

The Spiral model is a risk management strategy that combines the iterative development process model with aspects of the Waterfall approach. It allows for incremental product launches and refining as well as the possibility to develop prototypes at each point of the spiral. The model's most significant characteristic is its capacity to handle unexpected risks after the project has started, which is made possible by developing a prototype. Because it is a government-sponsored initiative, it requires adequate security and, if possible, some deliverables.

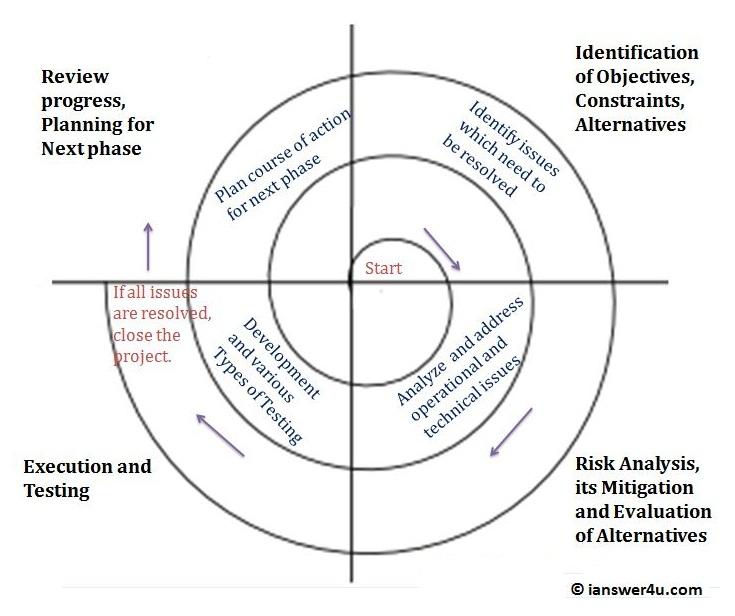


Figure 1: Project Process Lifecycle

## **Quality Gate for Each Phase of Software Development**

A quality gate for each step is essential for directing how each phase came out. The Spiral model was chosen as the project's model. Depending on the project risks, the project manager might adjust the number of phases required to produce the product. After selecting features, the product manager will assess and resolve risk concerns with the help of quality testers to ensure that the product being produced is up to par, and then it will be delivered. This procedure is repeated till the final product is given.

## **List of Tasks (Work Breakdown Structure)**

Figure 2: Work Breakdown Structure of the project

## **Estimation**

An estimation of each task using COCOMO81 model is given below-

Since this is an embedded system, the value of coefficient (c) is 3.6, project complexity (p) is 1.20, size (KLOC) is 315, T is 0.32

1. Effort= PM = c \* (size) P

         = 3.6 \*(315) 1.20

= 3583.23 person-month

1. Development Time= DM = 2.50 \* (Effort) T

      = 2.50 \* (3583.23) 0.32

           = 34.30 Months

~ 137 Weeks

1. Required number of people = ST = Effort / DM

       = 3583.23 / 34.30

               = 104.46 ~ 105 Persons

## **Schedule the Tasks**

|  |  |  |
| --- | --- | --- |
| **Tasks** | **Task Duration (Weeks)** | **Predecessor** |
| 1. Feasibility Study | 3 |  |
| 1. Defining Business Objectives | 3 | 1 |
| 1. Defining Requirements | 4 | 1, 2 |
| 1. Effort Estimation | 15 | 1, 2 |
| 1. Strategic Planning | 10 | 3, 4 |
| 1. Analyzing Historical Data | 5 | 5 |
| 1. Analyzing Complexity | 10 | 6 |
| 1. Analyzing All Risks | 10 | 6 |
| 1. Budget Estimation | 5 | 4, 8 |
| 1. Allocating Resource | 5 | 5, 9 |
| 1. Prototyping | 10 | 3, 10 |
| 1. Coding | 20 | 7, 11 |
| 1. Software Testing | 15 | 12 |
| 1. Installation | 10 | 13, 12 |
| 1. Client Training and Delivery | 5 | 14 |

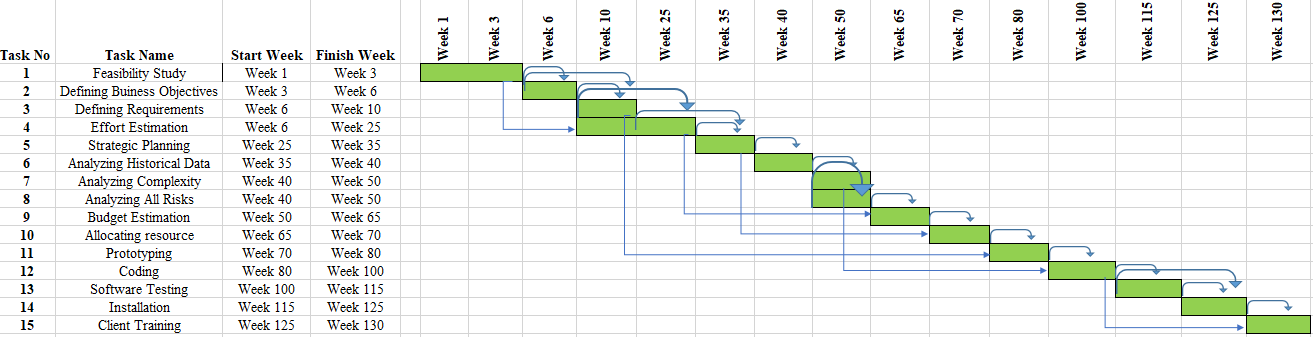


Figure 3: Gantt Chart of the Scheduled Tasks

**N.B.** In the task duration weekends are also included which will not be counted as working days. Team members and other employees will work 5 days a week and 8 hours a day.

## **Prepare List of Milestones**

|  |  |  |
| --- | --- | --- |
| **No** | **Item** | **Date** |
| 1 | Feasibility Study | May 23, 2021 |
| 2 | Defining Requirements | July 4, 2021 |
| 3 | Effort Estimation | November 28, 2021 |
| 4 | Strategic Planning | January 2, 2022 |
| 5 | Analyzing Complexity | April 3, 2022 |
| 6 | Analyzing All Risks | July 10, 2022 |
| 7 | Budget Estimation | August 14, 2022 |
| 8 | Allocating Resource | September 18, 2022 |
| 9 | Prototyping | October 30, 2022 |
| 10 | Coding | January 15, 2023 |
| 11 | Software Testing | June 28, 2023 |
| 12 | Installation | September 27, 2023 |
| 13 | Client Training and Delivery | December 17, 2023 |

## 

## **Staffing Plan**

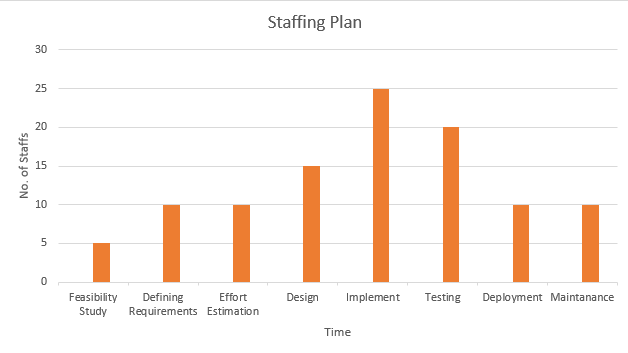


Figure 4: Staffing Plan

|  |  |  |
| --- | --- | --- |
| **Person** | **Assignment** | **Backup** |
| MD. Mustafizur Rahman | Project Manager | **Shadman Latif** |
| Shadman Latif | Functional Manager | MD. Mustafizur Rahman |
| MD. Sajadul Alam | Requirements Analyst | Rubiyet Fardous |
| Rubiyet Fardous | Technical Manager | Tasnim Haider |
| Tasnim Haider | Software Engineer | Mahir Afsar |
| Mahir Afsar | UI/UX Designer | Rubiyet Fardous |
| Shusmita Islam | Programmer | Marufuzzaman Rahat |
| Lamisa Yesmin | Tester | MD. Mustafizur Rahman |
| Marufuzzaman Rahat | Database Manager | Musfiqur Rahman |
| Musfiqur Rahman | Installation Engineer | Shusmita Islam |

## **Monitoring and Controlling Mechanism**

The monitoring and controlling component is the most important thing for any work. It ought to be well characterized. Weekly project meeting will take place under lead project manager’s supervision. The group pioneers will evaluate the circumstance day by day and report it in terms of announcing and recording the advance. The extend supervisors will do a month-to-month report. Other than this, each report will be form-based, where the advance will be kept on track based on a rate value together with all the desired qualifications. They can utilize and administration apparatuses for this.

Typically, one of the superior errands to imagine the advance for the venture directors. The Gantt chart can be utilized in arrange to do this so they can visualize the issues inside effectively. In case, in case the venture goes off track for any reason, and the advance slacks behind the plan, they can take quick activity for this. Other group individuals can have gotten to so that in case required, individuals from other parts of the module can be pulled in. Utilizing this procedure, the advance can be amended. On the other hand, they can contract transitory individuals to go back on track.

## **Risk Management**

It is way better to be secure than too bad. In this case, the objective is to be proactive instead of being reactive. There can be numerous risks included in this project.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S/N** | **Risk Description** | **Probability** | **Impact** | **Mitigation Plan** |
| 1 | Unrealistic time estimate | 60% | High | Take multiple estimation. |
| 2 | Loss of work due to equipment failure/loss | 30% | Significant | Weekly data backup to Hard drive. |
| 3 | Unavailability of API’s | 20% | Moderate | Alternative API’s will be checked for. |
| 4 | Developers needs to hardware or software requirements | 5% | Low | Select best available hardware or software components. |
| 5 | Exceeding budget | 15% | Moderate | Some extra budget needs to be added. |
| 6 | Testing and debugging error | 10% | Moderate | Adopting qualitative testers. |
| 7 | Failure of server | 8% | Low | Backup system database regularly. |
| 8 | Staff/Personnel shortfall | 5% | Low | Take some extra members in the team. |

## **List of Deliverables**

The list of deliverables of the project include:

* Project Plan
* Requirements Specifications
* First Demo Prototype (Design document)
* Second Demo Prototype (Program design)
* Test Manuals
* Final Product and User guides

## **Defect Tracking Process**

Defects are an exemption that's to be acknowledged. The defects are to be followed and reported. The report will be submitted from the tester or the developer to the test supervisor who at that point will continuously record the abandons. The abandons can be recorded when models or deliverables are at hand, or possibly indeed amid the development stage. The abandons are to be found, at that point categorized, settled by the engineers, tried by the analyzers, check whether the deformity can be brought to closure or not, report the imperfection.

A few precautionary estimations are ought to be taken to track defects.

* While the coding stage begins extended supervisor will continuously check the usage are based on legitimate requirements.
* Requirements outlined by client partners ought to be kept up and overhauled on a normal basis.
* There ought to be a palatable number of intuitive between the development team and the project manager.

## **Metrics**

Multiple types of metrics have been collected during the project duration.

* Productivity Metrics
* Gross Profit Margin
* Quality and Satisfaction
* Actual Cost
* Return on Investment
* Number of error and customer complaints
* Planned hour and actual time
* Time and budget change

## **Post Mortem**

The main goal of this project was to create a Software Development Project Management plan for Dhaka Subway system. A well-documented project plan was created and according to that plan the project was started.

All projects must drop inside the limits of success or failure. The lessons to be learned from them are the foremost vital stage. One of the few things which could’ve been arranged in an unexpected way or seem posture to be an issue is composed approximately. Selecting the process model was little bit challenging. Looking at the requirements Spiral model was chosen which combines the iterative development process model with aspects of the Waterfall approach. The model's most significant characteristic is its capacity to handle unexpected risks after the project has started, which is made possible by developing a prototype. That also, having a slight variance in it. The effort estimation was almost correct. Some changes came in scheduling task. Some tasks were completed before the scheduled date and some after the schedule. Some extra team members were needed because of unavailabity of some regular members.

Generally, the extend length may be a subject of the address of whether it could’ve been done at a prior date or not. Indeed, at that point, the estimations taken may demonstrate to be off-base and in general drop indeed past the given date.

Though some plans didn’t go according to the initial plan but almost works were done according to the plan and project has been completed successfully. The good and bad practices have been documented in the project document so that in near future project the best practices can be applied and bad practices can be avoided.