AMERICAN INTERNATIONAL UNIVERSITY- BANGLADESH

Department: Faculty Of Science & Technology

Project Title: **Software Development Project Management Plan for Dhaka Subway Systems Automated Ticket Issuing System.**

Course Tittle: **Software Development Project Management**

Section: **C**

Group Member Information:

Name	ID
1. **Dilruba Khanam Jesey**	**19-39979-1**
2. **Tareq Uddin Bhuiyan**	**18-37979-2**
3. **Ridwan Mannan Rahat**	**18-38990-3**
4. **Mirza Mohammad Samiul**	**17-33822-1**

Revision History Page :

Revision	Authors	Description	Date
Project Management Plan V.1.0	Dilruba Khanam Jesey	Initial creation of the system	11/09/21
Adding Features Plan V.1.1	Tareq Uddin Bhuiyan	Planning for adding features in the system	02/01/22
Editing Work V.1.2	Tareq Uddin Bhuiyan	Formal editorial work	13/01/22
Revise Features Plan V.2.0	Ridwan Mannan Rahat	Revising and minor changes in features plan	23/01/22
Resource Allocation V.2.1	Samiul Mirza	Adding/removing or restoring resources	31/01/22

Introduction:

This is the Dhaka Subway Systems Automated Ticket Issuing System software development project management plan. The software development lifecycle is explained in detail in this software project management plan. This document will contain detailed information on the project's management plan. The people in the IT department are the intended audience for this project. It specifies the technical and managerial approaches to software product development. All engineering and management activities that need to be transferred to the Dhaka metro system can be done so. This includes some variables that may have an impact on the project.

Process Model:

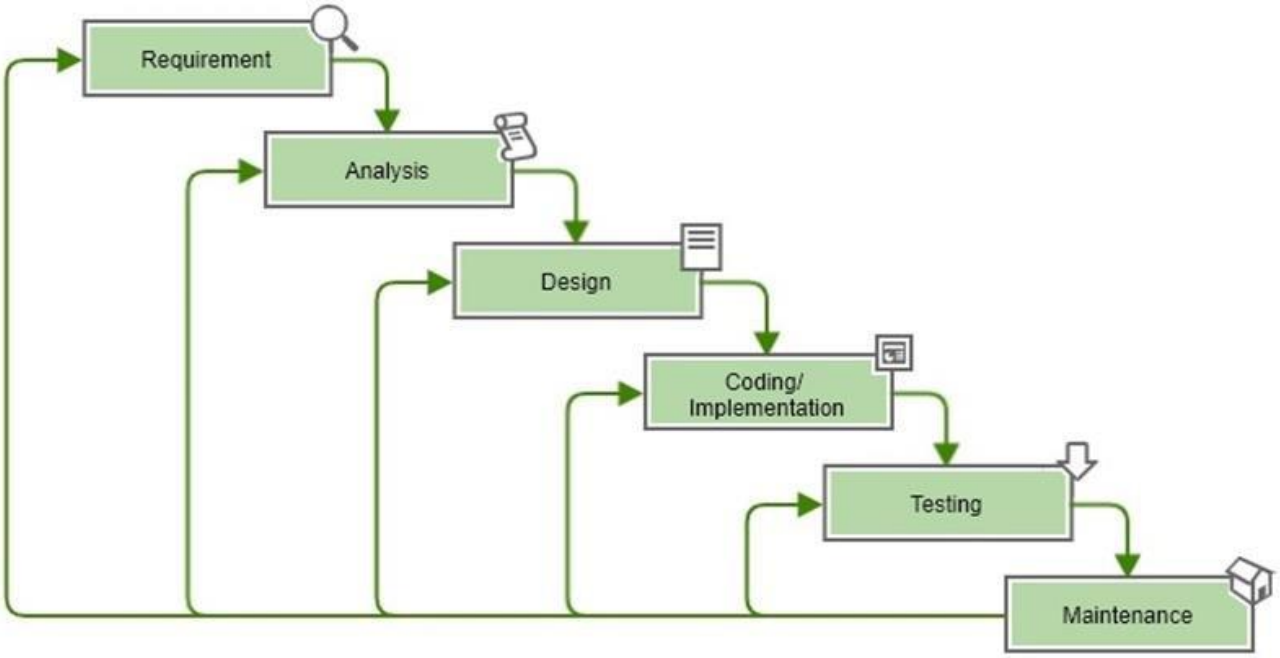
(a)As we all know, we need to deliver software that will allow Dhaka Subway Systems to create an automated ticketing system. Furthermore, our requirements are specific and well-defined. As a result, we'll use the waterfall model.

(b)The waterfall model is divided into several stages. So, in this model, there will be some phases, and each phase must be completed before moving on to the next. We must perform a quality assurance test before completing each phase. The documentation is extensive even before the start of each phase. If the client wants to make a change, they can do so during the development process.

Why This Model:

The Waterfall approach was the first SDLC model to be widely used in Software Engineering to ensure project
success.The entire software development process is divided into separate phases in "The Waterfall" approach. Typically, the output of one phase serves as the input for the next
phase in this Waterfall model.

The different phases of the Waterfall Model are depicted in the following diagram.



The sequential phases in Waterfall model are --

Requirement Gathering and analysis – This phase captures all possible requirements for the system to be developed and documents them in a requirement specification document.

System Design – This phase examines the requirements specifications from the previous phase and prepares the system design. This system design aids in defining the overall system architecture as well as specifying hardware and system requirements.

Implementation – The system is first developed in small programs called units, which are then integrated in the next phase, using inputs from the system design. Unit testing is the process of developing and testing each unit for its functionality.

Integration and Testing –After each unit has been tested, all of the units developed during the implementation phase are integrated into a system. The entire system is then tested for any faults or failures after it has been integrated.

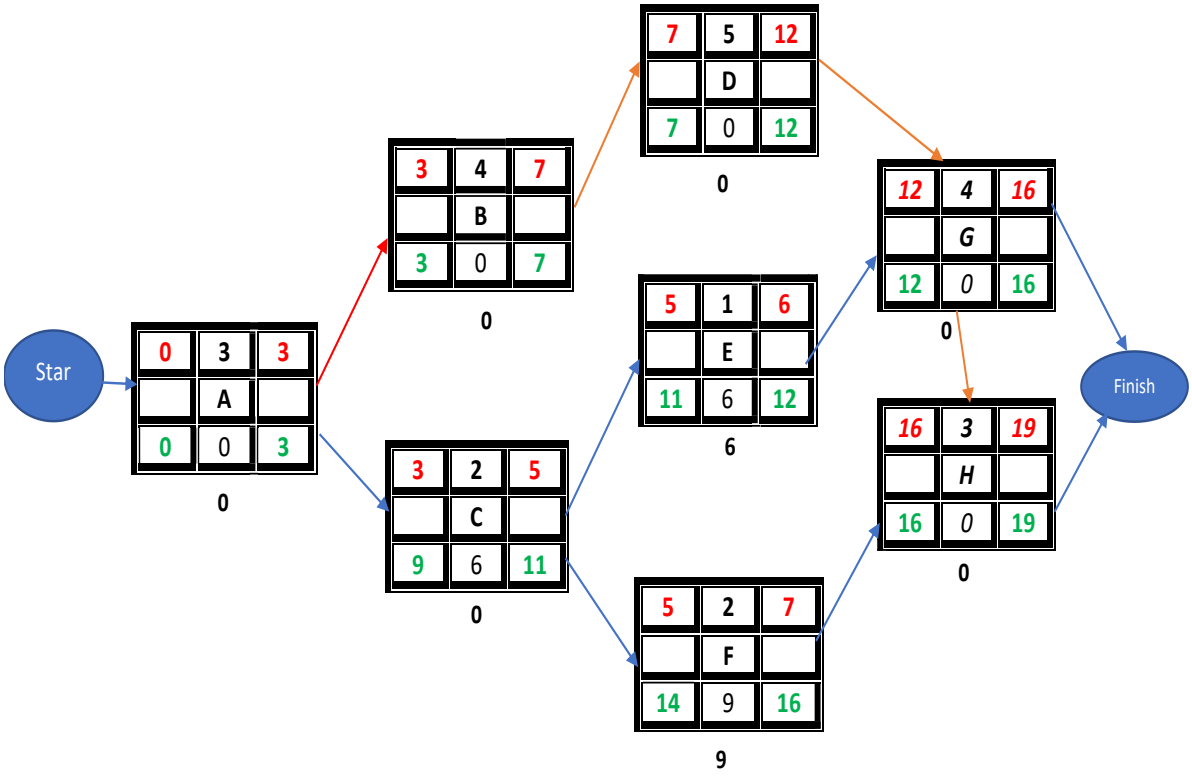
Deployment of system – The product is deployed in the customer environment or released into the market after functional and non-functional testing is completed.

Maintenance – In the client environment, there are a few issues that arise. Patches are released to address these issues. In order to improve the product, newer versions have been released. Maintenance is carried out in order to bring about these changes in the customer's environment.

The table below shows a list of tasks that are commonly used in software development projects:

Working Product	**QA Method**
Statement Of Work	Inspection
SDPM Planning	Inspection
Scheduling	Inspection
Software Requirements Specification	Inspection
System Design Document	Inspection
Code Implementation	Run

List of tasks (work breakdown structure, WBS):



Estimation (Use the COCOMO81 model):

COCOMO81 stands for Constructive Cost Estimation Model, which was created by Dr. Berry Boehm in 1981, hence the name. Using this method, we can estimate the cost of software.

COCMO81 is calculated using the following equation:

$\text{Effort} = c \times (\text{size})^k$

$\text{Development Time} = 2.5 \times (\text{effort})^t$

COCMO81 applies three class of software projects.

- Organic Project
- Semi-Detached Project
- Embedded Project

Software development projects can be classified into three types based on their complexity, according to COCOMO.

System Type	c	k	t
Organic	2.4	1.05	0.38
Semi-detached	3.0	1.12	0.35
Embedded	3.6	1.20	0.32

Our project will have 300000 lines of code, we estimate. Now we'll calculate the effort and time for each of the three software project classes.

Here, LOC = 300000

Therefore, KLOC=300

Organic:

Effort = c (KLOC) ^{k}

= 2.4(300) $^{1.05}$

= 957 person-month

Development time = 2.5(Effort)^{t}

= 2.5(957) $^{0.38}$

= 33.9 months

Semi-detached:

Effort = c (KLOC) ^{k}

= 3(300) $^{1.12}$

= 1784

Development time = 2.5(Effort) ^{t}

= 2.5(1784) $^{0.35}$

= 34.35 months

Embedded:

Effort = c (KLOC) ^{k}

= 3.6(300) $^{1.2}$

= 3379

Development time = 2.5(Effort) ^{t}

= 2.5(3379) $^{0.32}$

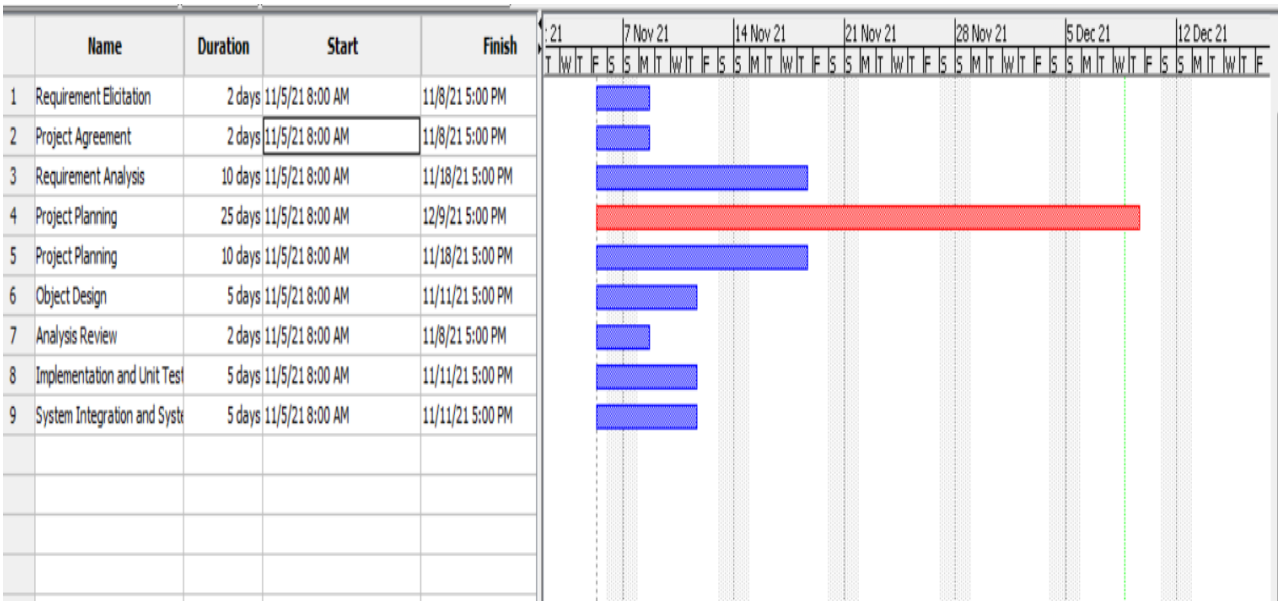
= 33.6 months

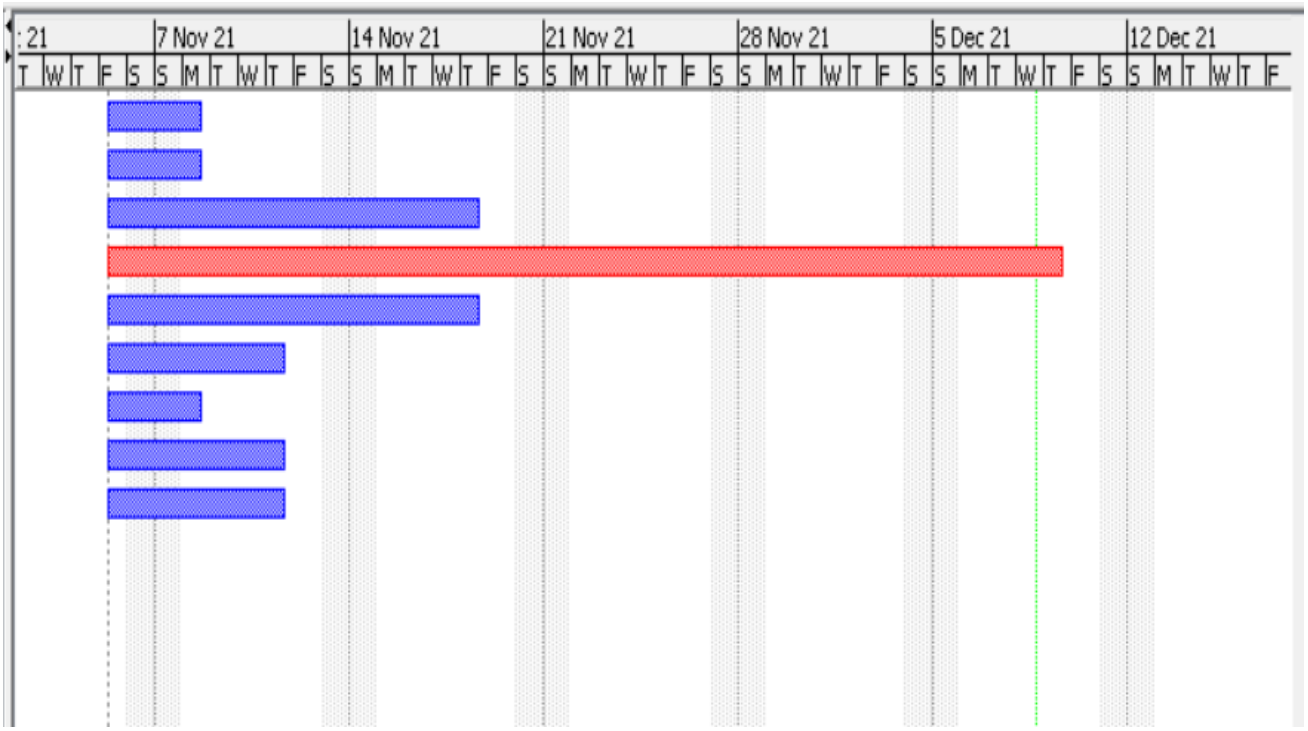
Schedule the Tasks:

Tasks	**Task Duration (Month)**	**Prerequisites**
Requirement Elicitation (a)	1	N/A
Project Agreement (b)	2	N/A
Requirement Analysis (c)	3	N/A
Project Planning (d)	4	b
System Design (e)	5	d
Object Design (f)	5	e
Analysis Review (g)	2	e,f
Implementation and Unit Testing (h)	5	e,f
System Integration and System Testing (i)	5	h

Gantt Chart for Following Schedule:

	Name	Duration	Start	Finish
1	Requirement Elicitation	2 days	11/5/21 8:00 AM	11/8/21 5:00 PM
2	Project Agreement	2 days	11/5/21 8:00 AM	11/8/21 5:00 PM
3	Requirement Analysis	10 days	11/5/21 8:00 AM	11/18/21 5:00 PM
4	Project Planning	25 days	11/5/21 8:00 AM	12/9/21 5:00 PM
5	Project Planning	10 days	11/5/21 8:00 AM	11/18/21 5:00 PM
6	Object Design	5 days	11/5/21 8:00 AM	11/11/21 5:00 PM
7	Analysis Review	2 days	11/5/21 8:00 AM	11/8/21 5:00 PM
8	Implementation and Unit Test	5 days	11/5/21 8:00 AM	11/11/21 5:00 PM
9	System Integration and Syste	5 days	11/5/21 8:00 AM	11/11/21 5:00 PM





List of milestones:

NO	Item	Milestone Date
1	Requirement Analysis and plan the phases	November 10, 2021
2	Risk Analysis and backup planes	November 25, 2021
3	Design	December 1, 2021
4	Coding	December 25, 2021
5	Testing/review	January 5, 2022
6	Deployment/Publish	January 25, 2022

Staffing Plan:

Person	**Assignment**	**Back Up**
Dilruba Khanam Jesey	Project Manager	Mirza Samiul
Tareq Uddin Bhuiyan	Technical Lead	Dilruba Khanam
Tareq Uddin Bhuiyan	UX Designer	Ridwan Mannan
Mirza Samiul	Developer	Tareq Uddin Bhuiyan
Ridwan Mannan	Data analyst	Diluba Khanam
Ridwan Mannan	Tester	Tareq Uddin Bhuiyan

Monitoring and Controlling Mechanism:

The project manager keeps track of the work's progress. The manager's room hosts a weekly or biweekly meeting. The project manager creates an agenda for the meeting before it takes place. Before the meeting, team members can suggest an additional agenda. During this meeting, the project manager assigned tasks. The project manager meets with the quality assurance manager during this meeting. The project manager meets with the team members to discuss the project. The project manager creates a detailed plan for the project. The project manager prepares a progress report.

Risk Management:

Surveillance: The project's features are mostly used with personal information such as credit card numbers, destinations, phone numbers, and other personal details. If any of this information is leaked, the users will be put in grave danger. It can be avoided by developing a high-security system in which only the authorized person has access to their own information and no one else.

Defect in planning: If any errors in the project plan are discovered during the project, it may take longer.

Communication Gaps: There is the potential for communication gaps between the team and its supporters. It's possible that if this happens, you'll end up falling behind. If this occurs, we will make a reasonable effort to locate that colleague. If they become unreachable for an extended period of time. The works will be rearranged.

Financial Crisis: The majority of the features must be adaptable to the needs of the client. As a result, it appears that the previous planning budget failed, necessitating the creation of a new budget. This resulted in a financial crisis. To avoid this issue, an expert can be hired to review a project's technical plans or cost estimate in order to boost confidence in the plan and lower project risk.

Design of the application and system: The consequences of developing the incorrect software functions, design, or architecture can be disastrous. As with technological risks, it is critical that the team includes experts who are familiar with the architecture and can make sound design decisions.

Performance: Any risk management plan should ensure that the user and partner expectations on performance are bound. Throughout the project, a standard and threshold testing must be considered to ensure that the work products are progressing in the right direction.

List of deliverables:

The technical and top-notch managerial processes that are critical for the development and deployment of the following system have been referred to in the Software Project Management plan:

Item	Description	Date
Software Requirement Specification	A software requirements specification, or SRS, is a document that describes how a system should function.	November 10, 2021
Risk Analysis	Risk analysis is a technique for identifying software risks.	November 15, 2021
Software Development Project Management Planning	A software development project management plan is primarily a schedule for the project. It specifies the start and end dates for each phase of the software development project..	November 20, 2021
Software Test Planning	A software test plan is a document that outlines the steps involved in software testing.	November 25, 2021
Software Quality Assurance Planning	The tools used to ensure that a product meets the standards specified in the software requirements specification are described in the Software Quality Assurance Plan.	January 1, 2022
Software Verification	Verification is the process of examining a program to determine if it complies with the requirements.	January 12-15, 2022

Defect Tracking Process:

The bug tracking process is referred to as the defect tracking process. We have a variety of tools for detecting bugs. However, we will use JIRA SOFTWARE in this project. This software serves as both a bug tracker and a project management tool. Bugs can also be found, recorded, and tracked. It also has a system that assigns the appropriate person to the appropriate time.

- JIRA is a popular project management tool among developers. This is widely used in industries because the methodologies interfaces are appealing to developers and testers, and they are simple to use. It is highly customizable, and it caters to a variety of users.

- JIRA, like other software, has some drawbacks. For example, the Agile method is well-suited to this software because it fully supports the Agile method. As we are using the Waterfall Model, it will be a little confusing. It has a confusing user interface, a file upload limit, and even complicated migrations.

- We'd categorize all of the divided parts as dangerous, moderate, or low so that we could easily give our best effort based on their severity.

- Before we start working on the project, we'll make sure it meets the needs of the Dhaka subway stakeholders.

- Following the completion of any phase of the product, we'll double-check to see if it meets the needs of the subway stakeholders.

- When it comes to communicating with the project manager and the subway's stakeholders, our team should be proactive.

List of matrices:

- Cost Matrix,
- Quality Matrix,
- Time Matrix,
- Code
- Matrix of Tests
- Productivity of Developers

Post Mortem:

The waterfall model is used to plan the overall project. A graphical user interface, a functional prototype, and a system integration prototype will be delivered. Before beginning Project Planning, an analysis will be conducted. The Object Design comes after the System Design. Both implementation and unit testing are expected to run concurrently. Unit Testing will be followed by System Integration. System testing will begin as soon as the system has been integrated. We are hopeful that the project will be completed on schedule. From start to finish, we will conduct a review of all phases and call a meeting whenever we achieve any major goal. For example, after gathering all requirements, completing the design, implementing the code, system testing, software quality assurance, and planning project deliverables, we will inspect each step. For the betterment of our project, we will follow best practices such as being clearly briefed on project scope and objectives, deliverables, and setting up keys to achieving milestones; each member will communicate with clarity and consistency; and we will identify all sensitive issues that may arise as a project risk and take the appropriate action to mitigate it. A sign-off meeting will be held.