PROJECT MANAGEMENT PLAN TEMPLATE

(Group No:09)

PROJECT MANAGEMENT PLAN

< Transport Management System Using QR Payment >

NAME	ID
Jannatuil Ferdouse Jannat	19-41735-3
A N M Rifat Wali Aditto	20-42391-1
Kazi Maksuda Rana Smrity	20-42351-1
Most.Afroja Mahomuda	20-43554-1

DATE

28-08-2023

TABLE OF CONTENTS

1.0 Introduction

The features and performance requirements of the software are described in this Software Requirements Specification (SRS) document. Additionally, it describes the features the product must have in order to meet the needs of all stakeholders. The expectations for connecting to other software and how users and other stakeholders will interact with the software are also illustrated in this SRS document. When writing the paper, consideration is given to interpersonal communication and real-world use. This SRS offers a thorough overview of the entire project. The development and testing teams may rely on it as a single source of truth. It acts as a course of action and keeps the development and maintenance teams informed of one another.

2.0 PROJECT MANAGEMENT APPROACH

Approaches to project management are organized procedures that direct the creation, implementation, monitoring, and conclusion of projects. To assure successful project delivery, these approaches offer a methodical manner to manage resources, time, scope, and other project variables. There are a number of popular project management strategies, each with its own set of guiding principles, procedures, and resources.

Roles and Organisational Structure:

Role in managing Project 1 and authority to make decisions:

In charge of the project's entire coordination, planning, execution, and monitoring. Additionally, decision-making authority has a significant amount of control on the project's goals, timeline, budget, and resource allocation.

Functioning Managers:

Position & Authority:

Manage several functional departments while providing subject matter expertise.

manage departmental operations, be a part of resource allocation choices, and ensure team availability.

Members of the project's team:

Function & Authority:

has cross-functional teams in charge of the tasks and deliverables for the project.tasked with doing certain tasks and sharing their expertise in their specialised domains.

Sponsor of the project and stakeholders

Role & Authority:

Individuals or organisations have an interest in the project's success.Make critical decisions, approve budgets, and establish the path of the project.

3.0 PROJECT TITLE: "Transport Management System Using QR Payment"

4.0 JUSTIFICATION:

A Transportation Management System (TMS) combined with a QR scanner has various advantages that justify its use. Here are some reasons why you should use a TMS with QR scanner functionality:

Efficient Tracking and Visibility: QR scanners allow for the tracking and viewing of cargo and vehicles in real time. This allows logistics managers to track the precise position, movement, and status of each consignment, which improves overall operational efficiency.

Accurate Data Capture: QR scanners capture data accurately and reliably, reducing the risk of manual errors associated with manual data entry. This precision ensures that the relevant cargoes are assigned to the appropriate vehicles and delivered to the appropriate locations.

Faster Processing: QR codes are swiftly scanned, saving time during the check-in, check-out, loading, and unloading processes. This speed helps to reduce supply chain wait times, congestion, and delays.

Enhanced Security: QR codes can be encrypted and validated, increasing cargo security. This prevents unauthorised access and lowers the likelihood of theft or tampering during travel.

Reduced Paperwork: The use of QR scanners to digitise procedures reduces the requirement for manual paperwork dramatically. This not only saves time and resources, but it also helps the environment.

5.0 OBJECTIVES AND PROJECT SCOPES:

Objectives:

- Create a user-friendly smartphone application for managing and booking transportation.
- Add QR code payment capabilities to facilitate simple cashless transactions.
- Use encryption and authentication techniques to ensure data security.
- Offer administrators real-time monitoring and reporting.

Scopes:

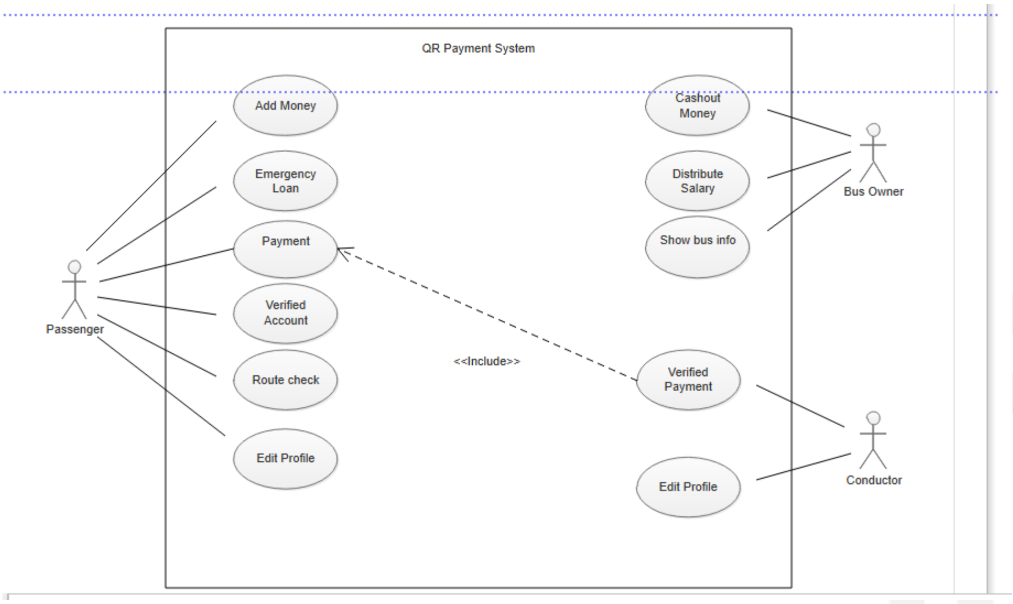
- Create and create the user interface for the mobile application.

- Use encryption and payment processing for QR codes.
- Establish a safe database to store user data and transaction history.
- Integrate with the APIs of external suppliers of transportation services.
- Create systems for user authentication and permission.
- Establish an administrative dashboard.

6.0 OVERVIEW OF THE PROJECT:

The Transport Management System using QR Code payment aims to improve transportation efficiency by integrating QR code technology with transport infrastructure. Passengers can scan QR codes on vehicles or stations, eliminating physical tickets and cash transactions. The system includes backend management tools for improved convenience, reduced transaction time, and better data insights**.**

Use Case Diagram:



7.0 STAKEHOLDERS ANALYSIS:

There are two types of stakeholders.

1. **Primary Stakeholders:** Primary stakeholders are people or groups who have a clear and important interest in how the project turns out.
   - Passengers
   - Conductor
   - Transportation Authorities
   - Technology Partners
   - Security and Privacy Promoter
2. **Secondary Stakeholders:** People or groups with a detached or less important interest in the project are secondary stakeholders.
   - Software Developers
   - Hardware providers
   - Insurance Companies
   - Consultants

8.0 MILESTONE LIST: Here's a milestone list for the development and implementation of a transport management system using QR code payment:

Milestone	Description	Date
Complete SRS	Define Transport Management System requirements for QR code payment, including user registration, trip booking, QR code generation, route optimization, and user profiles.	30.08.2023- 12.09.2023 (2 weeks)
Design	Design phase involves creating architectural and user interface design for a system, including database schema, components, technology stack, wire frames, QR code generation, scanning, and payment gateway integration.	13.09.2023- 03.10.2023 (3 weeks)
Complete Coding	Developers code system components, including user registration, authentication, trip booking, QR code generation, payment API integration, and route optimization.	04.10.2023- 31.10.2023 (4 weeks)
Complete Testing and Debugging	Coding and testing are crucial to identify and fix bugs, ensuring system stability, user flows, QR code payment processing, and database interactions.	01.11.2023- 07.11.2023 (7 Days)
Documents – User Guides and Installation	Prepare user guides and installation documentation for the Transport Management System, guiding users through booking trips, QR code payments, and system setup and deployment..	08.11.2023- 14.11.2023 (7 Days)

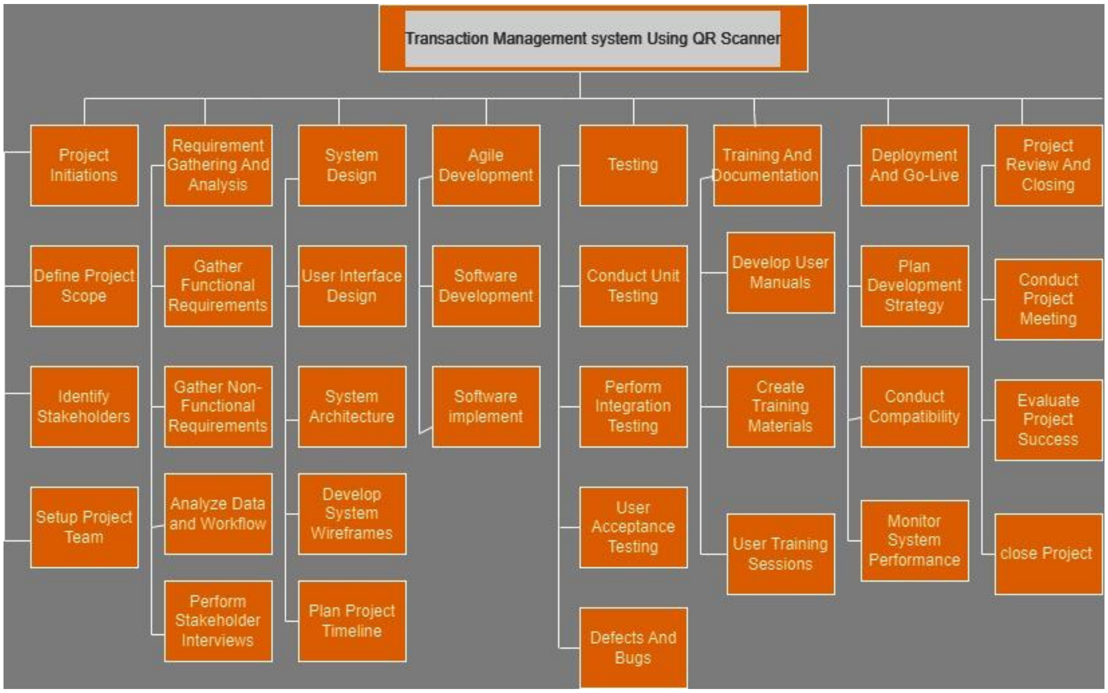
9.0 Process Model to be followed:

The process model chosen for designing a Transport Management System (TMS) with QR Code Payment is determined by several criteria, including project complexity, team size, requirement clarity, and the need for flexibility. The Agile methodology, specifically the Scrum framework, is one ideal process paradigm for this type of project.

Let us look at the reasons for this decision and the justification:

- **Iterative Development:** Creating a TMS with QR Code Payment necessitates the inclusion of complex functionality such as trip booking, payment processing, route optimisation, and user management. Agile development provides for iterative development, allowing you to add and refine features progressively. This is advantageous since you can obtain early customer input and swiftly adjust to changes.
- **Changing Requirements:** The QR code payment integration might involve third-party payment gateways, which could have evolving requirements or integration challenges. Agile methods are well-suited for accommodating changing requirements, as they emphasize regular communication with stakeholders and flexibility in adjusting the product backlog
- **User-Centric Design:** A TMS needs to be user-friendly and efficient. Agile methodologies prioritize regular user involvement and feedback through features like sprint reviews and daily stand-up meetings. This approach ensures that the developed system aligns with user needs and expectations.
- **Quick Time-to-Market:** The transportation industry can be competitive, and having a working product available fast might be advantageous. The incremental approach of Agile allows you to provide viable software at the end of each sprint, perhaps allowing for earlier release of software.
- **Risk Mitigation:** Creating a TMS with QR Code Payment entails technological concerns, such as QR code scanning and payment processing integration difficulties. Agile's short iterations and continuous testing assist in identifying and addressing risks early in the development process, minimising the likelihood of serious difficulties appearing later in the process.
- **Collaborations:**Agile approaches encourage cross-functional teams, including developers, testers, designers, and business stakeholders, to work closely together. This collaboration is critical in a TMS project to ensure that both the technical and functional components are well-integrated.
- **Transparency:** Scrum, a well-known Agile framework, establishes distinct roles (Product Owner, Scrum Master, Development Team) and ceremonies (Sprint Planning, Daily Stand-ups, Sprint Review, Sprint Retrospective). This openness improves communication and accountability among team members.
- **Continuous Improvement:**Agile methodologies foster continuous learning, enabling innovative features like QR code payments to adapt to user feedback.
- **Scalability:**If the TMS project expands or new features are required in the future, Agile frameworks such as Scrum can be scaled using practises such as Large-Scale Scrum (LeSS) or Scaled Agile Framework (SAFe).

10.0 WORK BREAKDOWN STRUCTURE:



11.0: ESTIMATION:

Effort Estimation :
Software Project Type: Semi-Detached
Effort Factor: 3.9/ 4.0
P (Productivity exponent): 2.09/ 1.09
T (Time exponent): 0.29/ 0.25
Source Lines of Code : 35,000
Effort Calculation:
Effort (PM) = Coefficient <Effort Factor> * (SLOC / 1000)^P
= 4 * (35,000 / 1000)^1.09
= 192.794 person-months
Development Time Calculation:
Development Time (DM) = 2.50 * (PM)^T
= 2.50 * (192.794)^0.25
= 9.315 months
Required Number of People Calculation:
Required Number of People (ST) = PM / DM
= 192.794 /9.315
= 20.697

Activity Diagram:

12.0 RESOURCE REQUIREMENTS

12.1 SOFTWARE REQUIREMENTS:

- Visual Studio, IntelliJ IDEA, or a comparable Integrated Development Environment (IDE).
- For code management and collaboration, use a version control system (e.g., Git).
- Web development programming languages (for example, Python, Java, JavaScript, and PHP).
- The TMS backend is built using a web framework (e.g., Django, Ruby on Rails, or Laravel).
- Front-end technologies (HTML, CSS, JavaScript) are used to create user interfaces.
- For data storage, use a database management system (such as MySQL or PostgreSQL).
- APIs or libraries for QR code production and scanning.
- APIs for payment gateway integration.
- Server and hosting infrastructure for deployment (e.g., AWS, Azure, Heroku).

12.2 HARDWARE REQUIREMENTS:

- Programmers' development computers (desktops/laptops) with sufficient processing power and memory.
- To ensure compatibility, use QA and testing devices such as smartphones, tablets, and laptops.
- TMS and associated databases are hosted on server infrastructure.

12.3 HUMAN RESOURCE REQUIREMENTS:

- **Project Manager:**The project manager is in charge of general planning, coordination, and making sure the project is completed on schedule and within budget. In addition to managing the project team, communicating with stakeholders, and reducing risks, the project manager.

- **UI/UX designer:**UI/UX designers are in charge of developing intuitive and aesthetically pleasing user interfaces for mobile applications. They put a lot of emphasis on user interface (UI) and user experience (UX) design.

- **Frontend developers:**These programmers create the user interface and put the UI/UX designers' ideas into practice. They are involved in the mobile application's client-side development.

- **Backend Developers:**Backend developers are in charge of creating the server-side logic, connecting APIs, putting payment processing into place, and maintaining data security.

- **Quality Assurance Testers:**QA testers make sure the application functions as intended and is error-free. They test using a range of gadgets, platforms, and circumstances.
- **Database Administrator:**If your project contains a complicated database structure, a database administrator may be required to effectively design and administer the database.
- **Security Specialist:**A security professional can assist put strong security measures in place to secure user data and payment information given the sensitive nature of payment data.
- **Technical Writers:**Technical writers can produce user manuals, training materials, and guides to aid users in navigating and comprehending the program.
- **Support and Training Personnel:**Personnel for User assistance and Training: Depending on the scope of the deployment, you may require personnel to offer user assistance and training both before and after the launch.
- **Business analysts and stakeholders:**These people will collaborate on the project to gather requirements, verify features, and make sure it satisfies business requirements.

13.0: PROJECT SCHEDULE:

Task name	**30august 2023**	**12sept 2023**	**13sept 2023**	**3oct 2023**	**4oct 2023**	**31oct 2023**	**1nov 2023**	**7nov 2023**	**8nov 2023**	**14 nov 2023**
Complete SRS	🟨	🟨								
Design			🟨	🟨						
Complete Coding					🟨	🟨				
Complete Testing and Debugging							🟨	🟨		
Documents – User Guides and Installation									🟨	🟨

14.0 DELIVERY PLAN:

Phase 1: Project Initiation (2 weeks)

Define the project's goals, boundaries, and stakeholders.

Assign roles to the project team when it is formed.

Make a communication strategy.

Create a comprehensive project schedule.

Phase 2: Conditions (6 weeks) Gathering and Design

Obtain specific needs from the relevant parties.

Create a user interface and a positive user experience (UI/UX).

Design the system architecture and database structure.

Select a payment gateway and integrate an external API.

Create prototypes and wireframes.

Phase 3: Integration and Development (10 weeks)

User interface implementation in front-end development.

Backend development: Create system logic and payment processing.

Integrate external APIs and payment gateways.

Put user authentication and permission into practice.

Set up security safeguards to protect data.

Phase 4: Testing and quality assurance

Unit-test each component individually.

Test the interoperability of various components.

User acceptability testing (UAT) should be done.

Address and resolve any problems or defects found.

Phase 5: Deployment and training (3 weeks)

Install the program on servers and app stores.

Offer user education and assistance materials.

first-round user testing and feedback gathering.

Phase 6: Final testing and refinement (3 weeks)

Conduct final testing and quality assurance inspections.

UI/UX should be improved depending on user input.

Take care of any difficulties that remain.

Phase 7: Activities for the launch and follow-up (2 weeks)

Publicly introduce the application in a formal manner.

Follow up on user comments and application performance.

Ongoing user assistance and post-launch problems should be addressed.

Phase 8: Documentation and Project Closure (1 week)

Conduct a review and assessment of the project.

Obtain input from team members and stakeholders.

Document and create a report on the project's completion.

Consider the lessons you've learnt for upcoming initiatives.

15.0 RISK ANALYSIS:

	Risks	**Probability**	**Impact**
1	System failure	30%	3
2	Late delivery	60%	5
3	Changes in requirements	40%	7
4	Poor Productivity	20%	9
5	Inadequate Management	80%	8

16.0 QUALITY CONTROL PLAN:

- **Quality Objectives:** Specify for the project precise, quantifiable quality objectives. This could include standards for performance, functionality, security, and user experience.
- **Quality Standards:** Specify the requirements for the project's deliverables' quality. Best practices for the industry, adherence to pertinent laws, and user expectations are a few examples of these standards.
- **Quality Assurance Activity:** Describe the procedures that will be followed to guarantee quality during the project's whole lifecycle:

  - Code Reviews: Regular code reviews help find and quickly fix problems.

  - Design review: Check the usability and alignment with user needs of the UI/UX design.

• Testing: Make plans for multiple testing phases, such as user acceptance testing, unit testing, integration testing, and security testing.

- **Roles and Responsibilities:** Describe the positions in charge of quality control:

• Project manager: Supervise how the Quality Control Plan is put into action.

• QA Testers: Perform testing and spot flaws.

• UI/UX Designers: Evaluate the usefulness and aesthetics of design elements.

- **Testing Procedures:** Describe the processes that will be used for the tests:

• Unit testing: Programmers check the accuracy of individual components.

• Integration testing: Examine how various components interact with one another.

• User Acceptance Testing: Participate end users in user acceptance testing to confirm functionality.

- **Defect Tracking and Resolution:** Define the procedures for finding, reporting, and fixing errors:

• To record and monitor issues, use a bug tracking system.

• Assign faults based on priority to the appropriate team members.

• Establish a procedure for checking and retesting defect fixes.

- **Documentation and Reporting:** Describe the process for recording and reporting quality-related information:

• Keep a log of all test cases, test outcomes, and issue reports.

• Produce consistent high-quality reports to inform stakeholders of your progress.

- **Approval Process:** Specify the standards and procedures for project deliverable approval:

• Specify the requirements that each deliverable must satisfy in order to be authorized.

• Describe how the necessary parties will be involved in the approval process.

17.0 BUDGET:

Name	Cost
Management & Implementation	**1 lakh**
Software Quality Testing	**1 lakh**
Maintenance	**50 thousand**
Hardware	**3.5lakh**
Utility	**1 lakh**
Total =	**= 7 lakh**

18.0 CONCLUSION:

The QR Payment initiative offers an innovative approach that combines cutting-edge technology with transportation services. This project intends to transform how consumers experience and pay for transportation while upholding the highest standards of security and quality by implementing QR code payments and a user-friendly interface.

References:

https://iopscience.iop.org/article/10.1088/1757-899X/590/1/012036/meta