

American International University-Bangladesh (AIUB) Department of Computer Science

Faculty of Science &Technology (FST) Spring 2022-2023

Section: A

Software Quality Assurance and Testing

Vehicle Management System

A Report Submitted By

|  |  |  |
| --- | --- | --- |
| SN | Student Name | Student ID |
| 1 | Nibal, Tasnim Alam | 20-42891-1 |
| 2 | Ahmed, Nasif | 20-42403-1 |
| 3 | Tumpa, Aditi Bhattacharjee | 19-41693-3 |

Under the supervision of

ABHIJIT BHOWMIK

Associate Professor & Special Assistant [OSA], Computer Science

Faculty of Science and Technology (FST),

American International University-Bangladesh

Software Test Plan

for

**Vehicle Management System**

Version 1.0 approved.

Prepared by

Tasnim Alam Nibal

Nasif Ahmed

Aditi Bhattacharjee Tumpa

American International University-Bangladesh (AIUB)

December 13, 2022

**Checked By Industry Personnel**

Name:

Designation:

Company:

Sign:

Date:

Table of Contents

[Revision History 3](#_Toc133851213)

[1. TEST PLAN IDENTIFIER: VMS0.1 4](#_Toc133851214)

[2. REFERENCES 4](#_Toc133851215)

[3. INTRODUCTION 4](#_Toc133851216)

[Background to the Problem 4](#_Toc133851217)

[Solution to the Problem 5](#_Toc133851218)

[4. REQUEIREMNT SPECIFICATION 5](#_Toc133851219)

[4.1 System Features 5](#_Toc133851220)

[4.2 System Quality Attributes 7](#_Toc133851221)

[4.3 System Interface 8](#_Toc133851222)

[4.4 Project Requirements 12](#_Toc133851223)

[5. FEATURES NOT TO BE TESTED 13](#_Toc133851224)

[6. TESTING APPROACH 13](#_Toc133851225)

[6.1 Testing Levels 13](#_Toc133851226)

[6.2 TEST TOOLS 14](#_Toc133851227)

[6.3 MEETINGS 15](#_Toc133851228)

[7. TEST CASES/TEST ITEMS 15](#_Toc133851229)

[8. ITEM PASS/FAIL CRITERIA 18](#_Toc133851230)

[9. TEST DELIVERABLES 18](#_Toc133851231)

[10. STAFFING AND TRAINING NEEDS 19](#_Toc133851232)

[11. RESPONSIBILITIES 19](#_Toc133851233)

[12. TESTING SCHEDULE 19](#_Toc133851234)

[13. PLANNING RISKS AND CONTINGENCIES 20](#_Toc133851235)

[14. APROVALS 21](#_Toc133851236)

Table of Figures

[Figure 1:Login Surface 8](file:///G:\SQT\final\Final%20Project\project\SQAT-Test-Plan-Final.docx#_Toc133981151)

[Figure 2:Action Page 9](#_Toc133981152)

[Figure 3: Modify Profile 9](file:///G:\SQT\final\Final%20Project\project\SQAT-Test-Plan-Final.docx#_Toc133981153)

[Figure 4: Add Vehicle 10](#_Toc133981154)

[Figure 5: Hire Vehicle. 10](#_Toc133981155)

[Figure 6: Profile Details. 11](file:///G:\SQT\final\Final%20Project\project\SQAT-Test-Plan-Final.docx#_Toc133981156)

[Figure 7: SOS Action. 11](file:///G:\SQT\final\Final%20Project\project\SQAT-Test-Plan-Final.docx#_Toc133981157)

[Figure 8: Selenium C# testing tool. 15](file:///G:\SQT\final\Final%20Project\project\SQAT-Test-Plan-Final.docx#_Toc133981158)

[Figure 9: Responsibilities. 21](file:///G:\SQT\final\Final%20Project\project\SQAT-Test-Plan-Final.docx#_Toc133981159)

[Figure 10: Time scheduling using MS Project 22](file:///G:\SQT\final\Final%20Project\project\SQAT-Test-Plan-Final.docx#_Toc133981160)

# Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| Revision | Date | Updated by | Update Comments |
| 0.1 | 2023.4.20 | Nasif Ahmed | First Draft |
| 0.2 | 2023.4.24 | Tasnim Alam Nibal | Second Draft |
| 0.3 | 2023.4.26 | Aditi Bhattacharjee Tumpa | Third Draft |

# TEST PLAN IDENTIFIER: VMS1.0

# REFERENCES

• https://iopscience.iop.org/article/10.1088/1757-899X/398/1/012014

# INTRODUCTION

The project is a web-based application designed to provide a platform for car owners, drivers, and admins to manage their daily activities related to vehicle hire and maintenance. The application is designed with a user-friendly interface and divided into three categories of users: car owners/general users, admins, and drivers. Each category has a set of features that cater to their specific needs. The car owner/general user can add and modify vehicle information, hire drivers, track vehicle information, and add emergency numbers. The admin has the power to delete or modify user information, approve or reject vehicle adding requests, and approve or reject driver profiles. The driver can check hire requests, set hiring rates, and see vehicle and car owner information. Overall, the project aims to provide a safe and efficient platform for vehicle hiring and management.

## Background to the Problem

* This project aims to address the increasing need for safe and reliable transportation services by providing a web-based platform for car owners, drivers, and users to connect and engage in a mutually beneficial way. With the rising population and urbanization, the demand for efficient transportation services has grown significantly, but finding trusted drivers and vehicles has remained a challenge. This platform will enable car owners to find reliable drivers, and users to find safe transportation services, while drivers can connect with potential clients and earn a steady income. The platform will also include features for emergency response and vehicle ownership management for administrators.
* The root cause of the problem is the lack of a convenient, safe, and efficient platform that connects car owners with drivers in need of work. This problem is significant because it addresses the safety concerns of car owners, the employment opportunities for drivers, and the need for a streamlined and user-friendly platform for both parties. Additionally, the problem of road safety and the prevalence of unlicensed and uninsured drivers highlights the need for a system that verifies drivers' identities and qualifications. By addressing these issues, the platform aims to provide a solution that benefits everyone involved while improving road safety and creating new job opportunities.

## Solution to the Problem

* The proposed solution is to develop a web-based platform that connects car owners with verified and professional drivers. It will offer features such as adding vehicle information, hiring drivers, searching for drivers, emergency alerts, ownership change, driver verification, and complaints against vehicle rule violations. This solution provides a reliable and efficient way for car owners to find verified and professional drivers and potential employment opportunities. It ensures safety and security by verifying the driver's identity and providing car owners with driver information before hiring. The solution can be a viable option for meeting the needs of both car owners and drivers due to its availability of resources, development costs, and user acceptance.
* The web application being specified is a car management system that serves as a platform for car owners, drivers, and administrators to manage their vehicles, drivers, and related information. The purpose of the software is to provide a convenient, reliable, and safe way for car owners to manage their vehicles, hire drivers, and keep track of important information such as emergency contacts and driver licenses. The system aims to streamline the process of car ownership and driver management, improve safety and security, and reduce administrative burdens for all parties involved. The software's key objectives and goals include providing a user-friendly interface, ensuring data accuracy and security, and enhancing communication and collaboration among car owners, drivers, and administrators.
* However, in general, there are several existing software solutions available to solve various problems. For instance, in the case of ridesharing and car hiring platforms, popular solutions include Uber, Pathao, and In-Drive. These platforms allow users to hire rides from drivers using mobile applications. Additionally, car rental platforms like Hiece-serve, Asho-rent-kori, and Enterprise provide rental cars for users. These solutions have revolutionized the way people hire cars and have made the process more convenient and accessible. However, each solution has its own strengths and weaknesses, and it is essential to evaluate them to determine the most suitable one for a specific problem. To solve those weakness and to overcome the problem of these renown software and mobile applications these do not have any solutions like web application and more developed that is why we developed our projects of Vehicle management system(VMS).

# REQUEIREMNT SPECIFICATION

## System Features

1.System Login

Functional Requirements

1.1 The software will allow users to login with their given username/mobile number and password.

1.1 If the entered username or password is incorrect more than five times, the system will issue a timeout of 30 minutes.

1.2 If the number of login attempts exceeds its limit (10 times), the system will block the user account and an email will be sent to the user's email address. After the verification, he will be able to login again. Priority Level: High

Precondition: The user has a valid user id/mobile phone and password

1. System Registration

Functional Requirements

* 1. At any time, any user can register in the system.
  2. The registering user will provide at least the necessary information to open an account.
  3. The registering user or user should only be able to register using their email address and password.
  4. After the successful registration, a confirmation email will be sent to the user's email.

Priority Level: High

Precondition: The user has a valid email and password

1. Update user information Functional Requirements
   1. Users can update their personal information any time.
   2. Users can change their passwords also.
   3. If any user forgets his password, he can get new password through email verification.

Priority Level: Medium

Precondition: The user must have an account.

1. Car user/general user’s Functional Requirements
   1. Users can add vehicle information, modify information, and add picture.
   2. Users hire driver, search for a driver, can see driver information and terminate driver for his vehicle.
   3. Users can Search vehicle with his number and can see some general information for safety and verification as well as driver information check is he valid driver or not.
   4. Users can add emergency number for provide alert to the registered number and a SOS button.
   5. Users can track some information about his own vehicle like there will be a note.
   6. Users can complain against any vehicle for violating rules using number plate.

Priority Level: High

Precondition: User has to login first.

1. Add/remove/check/charge/sos Functional Requirements
   1. Admin can delete any user and modify their information and view all details.
   2. Admin can Search vehicle with its number and can see all information of a vehicle.
   3. Admin can made Ownership change and can delete vehicle information from previous owner.
   4. Admin can approve or reject drive profile to continue as a driver.
   5. Admin will have a SOS button for emergency response.

Priority Level: Medium

Precondition: Admin has to login first.

1. Drivers Functional Requirements
   1. Driver can check hire request and can accept or reject.
   2. Driver can add and modify his information and add his driving license information.
   3. Driver can set his hiring rate for per hour and can set monthly salary.
   4. Driver can see all information of the car and car owner of that car which will be derived by him.
   5. Driver Can perform 3rd , 4th and 6th number feature of the user character.

Priority Level: High

Precondition: Driver has to login first.

## System Quality Attributes

**Usability:** The usability of the Vehicle Management System is an important aspect of its design. The user interface has been carefully crafted to be simple and user-friendly, ensuring that anyone can easily navigate and perform tasks. Users can easily view vehicle details, maintenance history, and generate reports. The searching option is accurate, ensuring that users can quickly find the information they need. The system loading time is fast, ensuring that users do not experience any delays while using the application. Additionally, the administrator has full access to the system, allowing them to manage and control the system with ease. Overall, the usability of the Vehicle Management System is optimized to provide users with a hassle-free experience.

**Maintainability**: The vehicle management system is built with maintainability in mind, ensuring that the maintenance team can easily perform their tasks. The system is designed to be stable, allowing for smooth operations without interruptions. Additionally, the system is easily adaptable, making it simple to make changes and updates as required. The team can quickly analyze the system to identify and fix any issues that may arise, ensuring that it runs efficiently at all times. Overall, the system's maintainability ensures that it can provide reliable service to users and be easily managed by the maintenance team.

**Efficiency**: The vehicle management system is designed with efficiency in mind. It is lightweight and optimized to consume less memory, allowing for a smooth and hassle-free experience for users. The system loading time is short, providing users with quick access to the information they need. This ensures that users can quickly perform necessary tasks without any delays.

**Reliability**: The vehicle management system is designed with high reliability in mind, ensuring that it can maintain its service provision under defined conditions for defined periods of time. The system is fault-tolerant, meaning it can withstand component failure and recover from any interruptions quickly. It has a robust database system that stores data securely and ensures that data is always accessible, even if there is a network or server failure. However, any potential downtime caused by server or network issues may lead to a temporary interruption in service.

**Functionality:** The vehicle management system's functionality is comprehensive and user-friendly. It includes all necessary features for managing vehicles, including adding new vehicles, tracking their location, and generating reports. The system also ensures data security by providing 100% protection for personal information and transactions. The high-security measures in place give users peace of mind and confidence in using the system for their vehicle management needs.

**Portability:** The vehicle management system is built with portability in mind, allowing it to be used in a variety of settings. It is compatible with different versions of Windows, making it accessible to a wide range of users. The system's adaptability allows it to be easily modified to suit different platforms or environments. Object-oriented design and implementation practices have been used in the system's development, which contributes to its portability. These practices enable the system to be easily transferred from one platform to another without requiring significant changes to its underlying code, thereby saving time and effort in development and maintenance.

## System Interface

1. Login surface of vehicle management system:

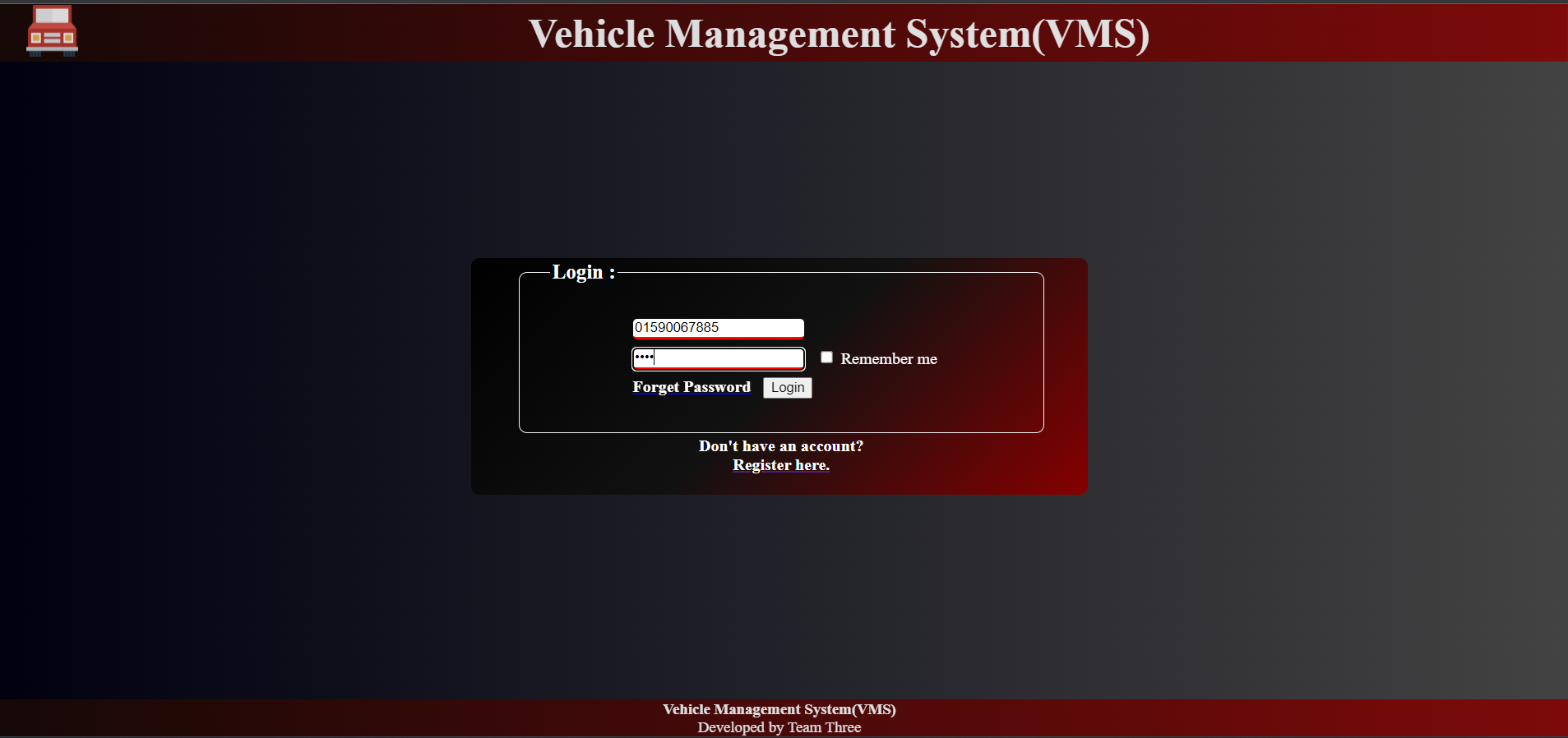


Figure 1:Login Surface

1. After login Action page of VMS:

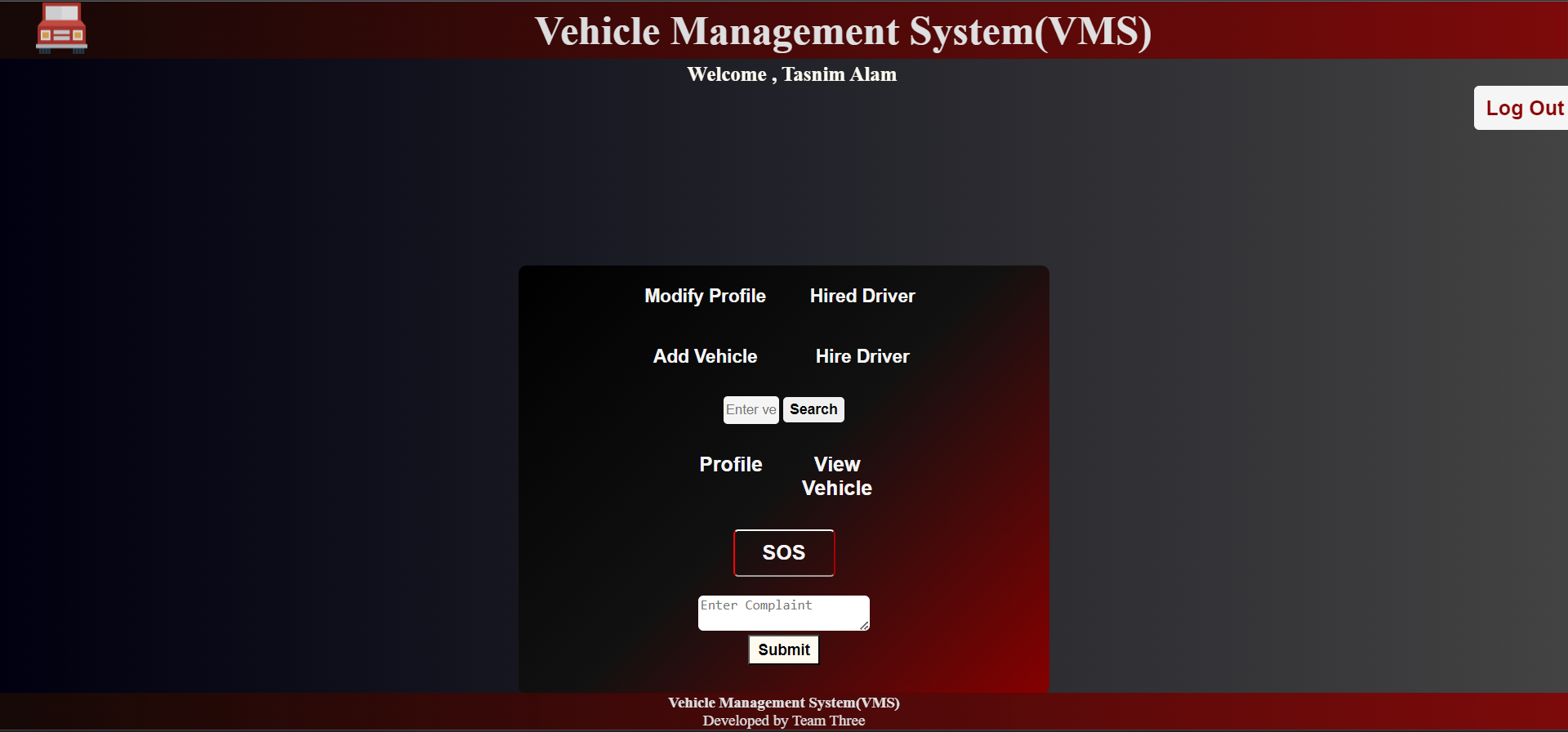


Figure 2:Action Page

1. A screenshot of a computer

   Description automatically generated with medium confidenceModify profile page:

Figure 3: Modify Profile

1. Add Vehicle Page:



Figure 4: Add Vehicle

1. Hire Vehicle Page:



Figure 5: Hire Vehicle.

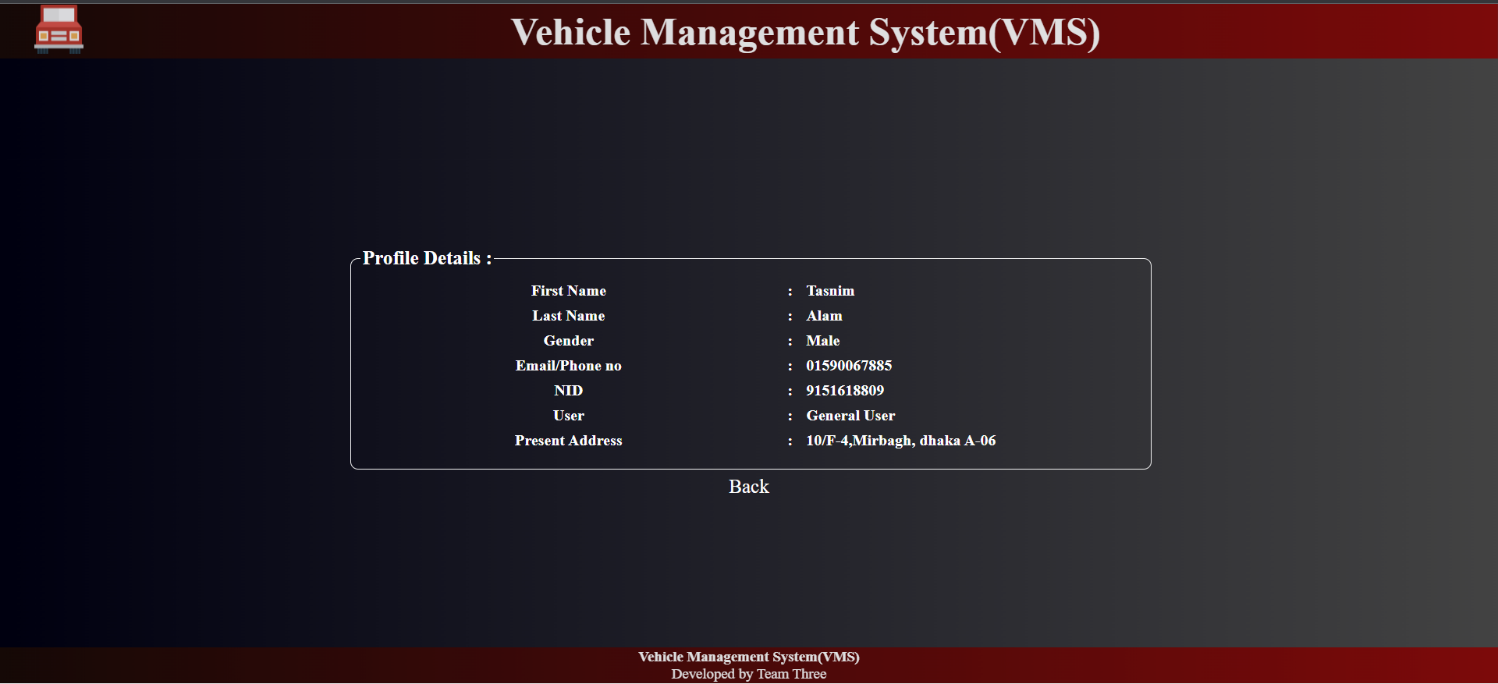
1. Profile Details Page:

Figure 6: Profile Details.

1. A screenshot of a computer

   Description automatically generated with medium confidenceSos(Emergency) Steps Page:

Figure 7: SOS Action.

## Project Requirements

**Budget Estimation:**

3 developers and engineers working for 2 months on a vehicle management system project:

Duration in weeks = 7 weeks

Office days = 5 days

Working hours = 8 hours

So, per week working hours is = (5 x 8) hours. = 40 hours

Hence, Total Working hours is = (40 x 8) hours. = 320 hours per developer

Developer salary is = 1200 BDT

Total developers Salary = (1200\*320) BDT

= 384,000 BDT

**Project Estimation**

COCOMO is a procedural cost estimate model for software projects and often used as a process of reliably predicting the various parameters associated with making a project such as size, effort, cost, time, and quality.

So, we will used Constructive Cost Model (COCOMO) to calculate the estimation of our project were,

D = Total time required for project development in Months (M). KLOC = The size of the code for the project in Kilo lines of code.

a, b, c, d = The constant parameters for a software project.

Consider, project is organic.

For, Organic Software Project, a= 2.4 b= 1.05 c=2.5 d= 0.38

Consider, kilo line of code KLOC=7000

**Effort Estimation Formulas:**

Effort, E = (a(KLOC)^b) \*EAF

= (2.4 \* (7000/1000) ^1.05) \* 0.81

= 27.94 ~28

= 28 person-months

Total Development Time, D = c(Effort)^d

= 2.5\*(28) ^0.38

= 9.89 ~10

= 10 months

Required number of people = Effort/Time

= 28/10

= 2.8 ~ 3

= 3 developers

Cost:

Labor cost = 3 \* 20,000 = 60,000/- Taka (Approximately)

Service cost = 5000\*3 = 15000/- Taka (Approximately) so,

Total cost = 60,000 + 15000 = 75,000/- Taka (Approximately)

# FEATURES NOT TO BE TESTED

In software testing for vehicle management system, there may be certain features or aspects that are not relevant or necessary to test for various reasons. Some examples of such features include:

• Third-party integrations: If the vehicle management system integrates with third-party software or hardware that is outside the scope of the project, then testing those integrations may not be necessary. For example, if the system integrates with a GPS tracking system that is already tested separately, there may not be a need to test it again as part of the vehicle management system.

• Outdated or deprecated functionality: If certain features of the vehicle management system have been deprecated or are no longer in use, it may not be necessary to test them. For example, if the system used to support an older version of a vehicle tracking device that is no longer used, then testing that functionality may not be relevant anymore.

• Design elements: Certain design elements of the vehicle management system, such as the colour scheme or fonts used, may not be necessary to test. These elements are generally cosmetic and do not affect the functionality of the system.

• Unlikely edge cases: Testing for every possible edge case and scenario can be time-consuming and impractical. Therefore, testing unlikely or rare edge cases that are unlikely to occur in real-world scenarios may not be necessary.

# TESTING APPROACH

## Testing Levels

* The testing strategy for the upcoming Vehicle Management System (VMS) project is comprised of three testing levels, including Unit, System/Integration (combined), and Acceptance tests. It is anticipated that there will be at least one dedicated full-time independent test person for system/integration testing. However, due to budgetary and time constraints, most of the testing will be conducted by the test manager with active participation from the development teams.
* UNIT Testing will be done by the developer and will be approved by the development team leader. Before Unit Testing is considered acceptable and passed on to the test person, the programmer must provide evidence of unit testing in the form of a test case list, sample output, data printouts, and defect information to the team leader. The test person must also be provided with all unit test information.
* SYSTEM/INTEGRATION Testing will be carried out by the test manager and the development team leader, with assistance from individual developers when needed. No specific test tools have been identified for this project. Programs will enter the System/Integration test phase only after all major defects have been rectified. Programs can have up to two major defects, provided that they do not hinder testing of the program, and a workaround is available for the error.
* ACCEPTANCE Testing will be performed by the end-users with assistance from the test manager and development team leader. The acceptance test will be conducted in parallel with the current manual ZIP/FAX process for one month after the completion of the System/Integration test process.

## TEST TOOLS

Our testing team has successfully tested the Vehicle Management System using Selenium with C#. Selenium is a popular open-source framework widely used for automated web application testing. Its integration with C# provides a powerful toolset for automated testing. Our team has utilized Selenium's features such as test recording, playback, and debugging to ensure comprehensive testing of the system. Through this testing approach, we were able to identify and fix various issues and ensure the system is functioning correctly. Selenium with C# has allowed us to streamline the testing process and improve the quality of the Vehicle Management System.

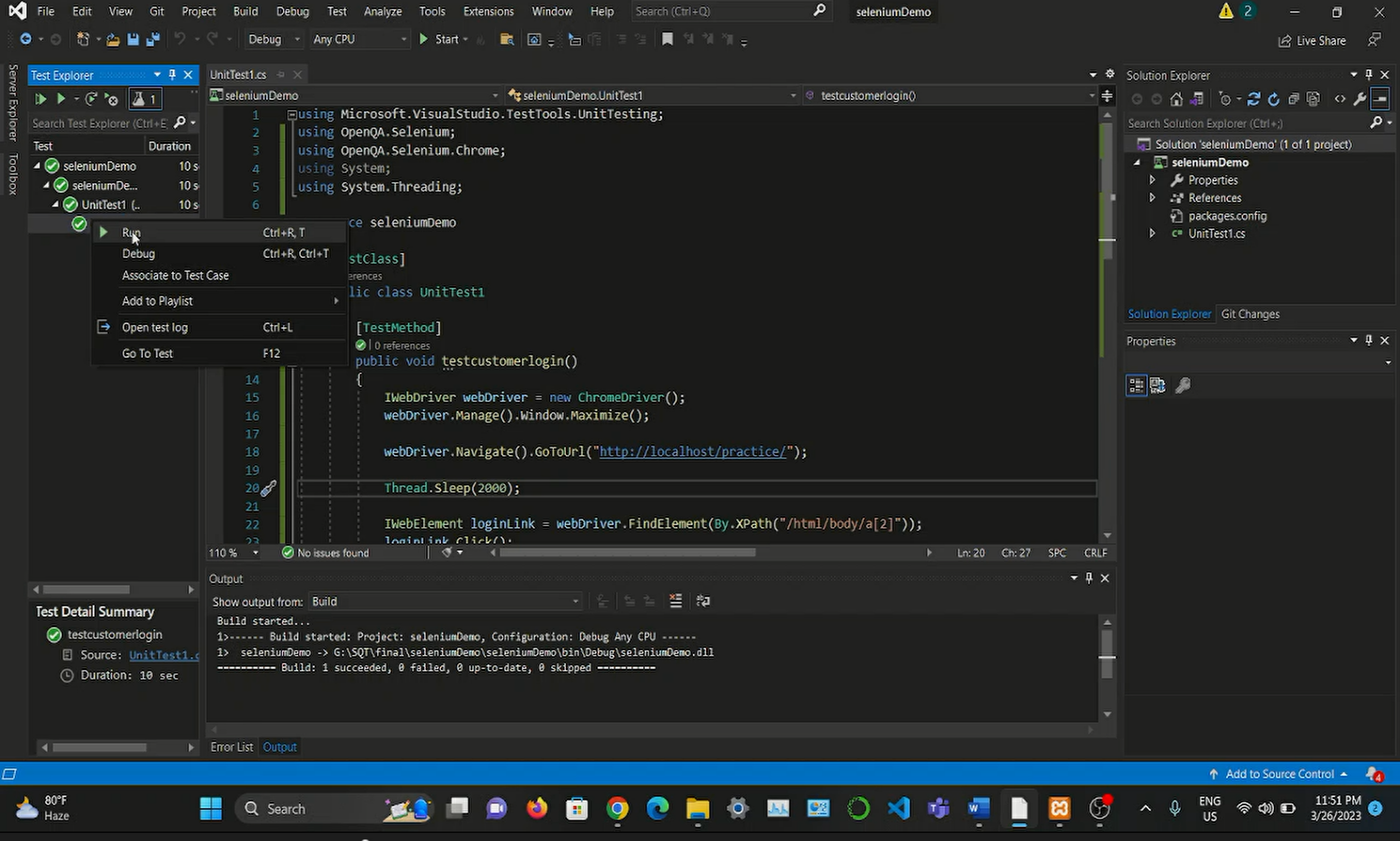


Figure 8: Selenium C# testing tool.

## MEETINGS

The test team will meet once every day to evaluate progress to date and to identify error trends and problems as early as possible. The test team leader will meet with development and the project manager once every day as well. Additional meetings can be called as required for emergency situations.

# TEST CASES/TEST ITEMS

(Example by some sample do it following our project features if any queries don’t hesitate to knock me via teams messengers phone calls #nibal )

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Project Name: Vehicle Management System (VMS) | | | Test Designed by: Nasif Ahmed | | |
| Test Case ID: OP\_1 | | | Test Designed date: 24-04-23 | | |
| Test Priority (Low, Medium, High): High | | | Test Executed by: Tasnim Alam Nibal | | |
| Module Name: Registration Session | | | Test Execution date: 25-04-23 | | |
| Test Title: Register user with First name, Last name, valid mail/Phone number, gender, password, nid, present address | | |  | | |
| Description: Test web application registration page | | |  | | |
| The precondition (If any): User must have valid mail/phone number and password | | | | | |
| Test Steps | Test Data | Expected Results | | Actual Results | Status  (Pass/Fail) |
| 1. Go to the website tap on the registration page  2. Enter both first and last name  3. Enter valid email/phone number  4. Select gender  5. Make a password  6. Enter NID number  7.enter present address | Enter first name: Alam  Last name: Nibal  Enter email/phone: 01743830469  Select gender: Male  Password: \*\*\*\*\*  NID: 245\*\*\*\*\*  Present address: 21/b | If the all information is correct then signup is done. | | As expected | Pass |
| Post Condition: User is validated with database and successfully registration to the account after successful account creation. The account session details are logged in the database. | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Project Name: Vehicle Management System (VMS) | | | Test Designed by: Aditi Bhattacharjee Tumpa | | |
| Test Case ID: OP\_2 | | | Test Designed date: 24-04-23 | | |
| Test Priority (Low, Medium, High): Medium | | | Test Executed by: Nasif Ahmed | | |
| Module Name: Login Session | | | Test Execution date: 25-04-23 | | |
| Test Title: Verify login with valid Username and Password | | |  | | |
| Description: Test web application login page | | |  | | |
| The precondition (If any): The user must have valid mail and password | | | | | |
| Test Steps | Test Data | Expected Results | | Actual Results | Status (Pass/Fail) |
| 1. Go to the website or application 2. Enter email/phone 3. Enter the password  4. Click login | Email/phone:  admin123@gmail.com    Password: hello123@ | Users should login to the application. | | As expected | Pass |
| Post Condition: The user is validated with the database and successfully login to the account. The account session details are logged in the database. | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Project Name: Vehicle Management System | | | Test Designed by: Tasnim Alam Nibal | | |
| Test Case ID: OP\_3 | | | Test Designed date: 24-04-23 | | |
| Test Priority (Low, Medium, High): Medium | | | Test Executed by: Aditi Bhattacharjee Tumpa | | |
| Module Name: User Information Update | | | Test Execution date: 2-04-23 | | |
| Test Title: Updates all the information for the users | | |  | | |
| Description: By using this function Updates all the information for the users and adds new info for the users | | |  | | |
| The precondition (If any): The user must have a log in the system | | | | | |
| Test Steps | Test Data | Expected Results | | Actual Results | Status (Pass/Fail) |
| 1,Go to the website  2.log in to the system  3.Give the user  Id  4.update Email.  5.update the password. 6. Click submit | User Id: 101 Email:  imoinul169@gmaul  .com  Password:321@@ | If all the information is correct then you can update the user information | | As expected | Pass |
| Post Condition: The user is validated with the database and successfully updating user information to the account. The account session details are logged in the database. | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Project Name: Vehicle Management System | | | Test Designed by: Tasnim Alam Nibal | | |
| Test Case ID: OP\_4 | | | Test Designed date: 24-04-23 | | |
| Test Priority (Low, Medium, High): Medium | | | Test Executed by: Nasif Ahmed | | |
| Module Name: Add/remove vehicles | | | Test Execution date: 25-04-23 | | |
| Test Title: Add or Remove medicine from the store | | |  | | |
| Description: Allocation of all vehicles | | |  | | |
| The precondition (If any): The admin must have a log in the system | | | | | |
| Test Steps | Test Data | Expected Results | | Actual Results | Status (Pass/Fail) |
| 1. Go to the website or application 2. log in with admin user 3. Give the user  Vehicle Id  4. Add new vehicle 5. Remove vehicle 6. Click submit | Log in with an admin account.  Vehicle Id: 101  Add new vehicle:  Jeep car  Remove vehicle: jeep 101 | At first login to the system. For adding the vehicles or remove the vehicles must give the medicines name | | As expected | Pass |
| Post Condition: The user is validated with the database and successfully adding or removing vehicles to the account. The account session details are logged in the database. | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Project Name: Vehicle Management System | | | Test Designed by: Tasnim Alam Nibal | | |
| Test Case ID: OP\_4 | | | Test Designed date: 24-04-23 | | |
| Test Priority (Low, Medium, High): High | | | Test Executed by: Nasif Ahmed | | |
| Module Name: Hire request acceptance | | | Test Execution date: 25-04-23 | | |
| Test Title: approves customer request and go for drive | | |  | | |
| Description: Hire approval to all hire request | | |  | | |
| The precondition (If any): The admin must have a log in the system | | | | | |
| Test Steps | Test Data | Expected Results | | Actual Results | Status (Pass/Fail) |
| 1. Go to the website or application 2. log in with Driver user 3. Let the user see how many requests he have.  4. then approve them | Log in with a Driver account.  Vehicle request: Jeep  Add new vehicle:  Jeep car  Remove vehicle: jeep 101 | At first login to the system. For adding the vehicles or remove the vehicles must give the medicines name | | As expected, | Fail |
| Post Condition: The user is validated with the database and approving driver requests to the account. The account session details are logged in the database. | | | | | |

# ITEM PASS/FAIL CRITERIA

The test process will be completed once the initial set of distributors have successfully sent in reassigned sales data for a period of one month and the new EDI data balances with the old ZIP/FAX data received in parallel. When the sales administration staff is satisfied that the data is correct the initial set of distributors will be set to active and all parallel stopped for those accounts.

# TEST DELIVERABLES

* Acceptance test plan
* System / Integration test plan
* Unit test plans/ turnover documentation
* Screen prototypes
* Report mock-ups
* Defect/ Incident reports and summaries
* Test logs and turnover reports

# STAFFING AND TRAINING NEEDS

It is great to hear that one full-time tester has been assigned for testing each unit and its integration capability. In addition, it is important to ensure that the testers and developers are trained on the basic operations of Selenium and Figma interface.

Selenium is an open-source testing tool used to automate web browsers, while Figma is a user interface design tool. Both tools are widely used in software testing and design, and it is important for the testers and developers to have a good understanding of their basic operations.

The training can be conducted through a combination of online resources, such as tutorials and videos, as well as practical hands-on training sessions. The training should cover topics such as how to create test cases, how to use test automation frameworks, and how to integrate test scripts into the development process.

Once the project has been accepted, it is also important to provide training to other staff members who will be involved in testing and maintaining the vehicle management system. This training should cover the basic Figma and Selenium testing process, as well as any specific testing requirements for the vehicle management system.

Overall, providing adequate training to the testers and developers on Selenium and Figma interface will help ensure the quality and reliability of the vehicle management system, and also help streamline the development and testing process.

# RESPONSIBILITIES

Figure 9: Responsibilities.

# TESTING SCHEDULE

The project plan for the vehicle management system includes specific time allocations for testing activities. These activities are detailed in the project timeline and plan, including the dates and times for each task and the personnel required for each process. The project manager will coordinate the test team, development team, management, and customer to ensure smooth execution of each task. To facilitate this, a project management tool will be utilized to create and maintain the schedule. By allocating time and resources for testing, the project team can ensure that the system is thoroughly tested and validated before being delivered to the customer.

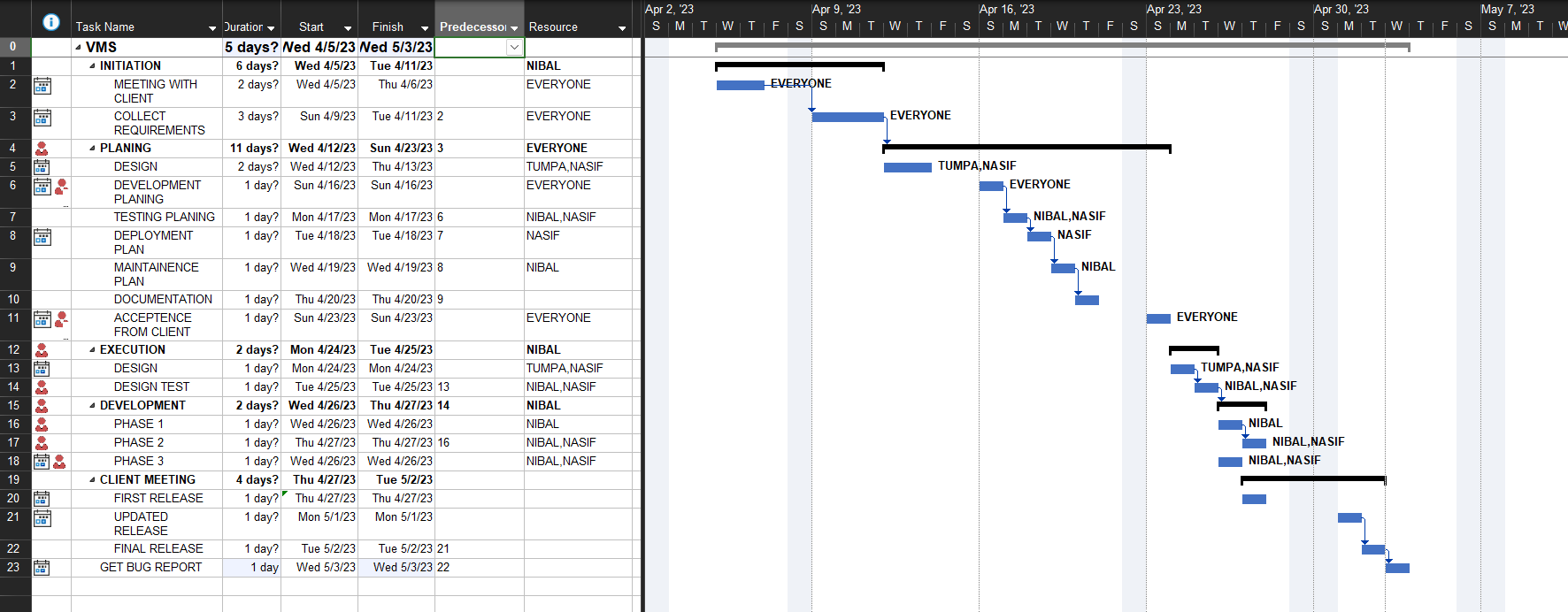


Figure 10: Time scheduling using MS Project

# PLANNING RISKS AND CONTINGENCIES

Effective risk management is crucial for any organization to minimize future losses. In order to manage risks, a proactive approach must be taken by corporate managers to identify, classify, and manage potential risks before actual execution of the project.

Risks can be classified into different categories such as

schedule,

operational,

technical,

programming,

budget,

and external risks.

For a vehicle management system, schedule risks may include imprecise time calculation, frequent project scope expansion, and misallocation of resources. Operational risks can arise from insufficient resources, lack of resource planning, poor communication within teams, improper task management, and failure to assign responsibilities. Technical risks could stem from continuous changes in requirements, complexity of product implementation, and improper integration of modules. By addressing all possible causes of risks, total risk cannot be eliminated, but it can be minimized for the vehicle management system.

# APROVALS

|  |  |
| --- | --- |
| Project Manager | APPROVED |
| Developer | APPROVED |
| Test Planner | APPROVED |
| Test Lead | APPROVED |
| Tester | APPROVED |
| End User | APPROVED |