

---

# **Software Test Description**

for

# **Workshop Management System**

**Version 1.0 approved**

**Prepared by**

**CB23116 AMALA KARTHIGAYAN**

**CB222141 NUR ALYA SYAKIRAH BINTI NASARUDIN**

**CB22139 VARSHINI JAGARAJAN**

**CB23101 NUR NABIHAH FATINY BINTI ISMAIL**

**20/6/2025**

## **Table of Contents**

1.0	Introduction	3
1.1	System Identification	3
1.2	References	4
2.0	Formal Qualification Test Preparation	5
2.1	Testing Schedule	5
2.2	Test Case of System	7
2.2.1	Manage Profile (Alya Syakirah – CB22141)	7
2.2.2	Manage Shop Inventory (Varshini Jagarajan – CB22139)	8
2.2.3	Request Inventory (Varshini Jagarajan – CB22139)	8
2.2.4	Manage Foreman Schedule (Amala Karthigayan – CB23116)	9
2.2.5	Select Available Schedule (Amala Karthigayan – CB23116)	9
2.2.6	Manage Payroll (Nabihah Fatiny – CB23101)	10
2.2.7	Manage Rating (Nabihah Fatiny – CB23101)	10
2.3	Hardware Preparation	11
2.4	Software Preparation	12
3.0	Formal Qualification Test Description	13
3.1	Manage Profile (Alya Syakirah – CB22141)	13
3.2	Manage Shop Inventory (Varshini Jagarajan – CB22139)	16
3.3	Request Inventory (Varshini Jagarajan – CB22139)	17
3.4	Manage Foreman Schedule (Amala Karthigayan – CB23116)	18
3.5	Select Available Schedule (Amala Karthigayan – CB23116)	22
3.6	Manage Payroll (Nabihah Fatiny – CB23101)	26
3.7	Manage Rating (Nabihah Fatiny – CB23101)	29
4.0	Acronym and Abbreviation	31

## 1.0 Introduction

### 1.1 System Identification

The Software Test Description (STD) belongs to the Workshop Management System.

System Title	Workshop Management System									
System Abbreviation	WMS									
System Identification Number	100 (SEP-SRS-WMS 1.1.25) The system identification number shows the identification number for Workshop Management System									
System Version Number	1.0 This is initial release number for Workshop Management System									
System Release Number	1.1.25 Start with number 1 as the major release number and middle with number 1 as the minor release number that this is initial version of the system. Number 25 as the service release that shows the system released in 2025									
Document Identification ID	SRS-WMS-2025 <table border="1"><tr><td>SRS</td><td>Software Requirement Specification</td></tr><tr><td>WMS</td><td>Workshop Management System</td></tr><tr><td>2025</td><td>Year of document created</td></tr></table>		SRS	Software Requirement Specification	WMS	Workshop Management System	2025	Year of document created		
SRS	Software Requirement Specification									
WMS	Workshop Management System									
2025	Year of document created									
Use Case ID	UC01-WMS-2025 <table border="1"><tr><td>UC</td><td>Use Case</td></tr><tr><td>01</td><td>Number of use case in the system</td></tr><tr><td>WMS</td><td>Workshop Management System</td></tr><tr><td>2025</td><td>Year of document created</td></tr></table>		UC	Use Case	01	Number of use case in the system	WMS	Workshop Management System	2025	Year of document created
UC	Use Case									
01	Number of use case in the system									
WMS	Workshop Management System									
2025	Year of document created									
Requirement ID	REQ-WMS-2025-101 <table border="1"><tr><td>REQ</td><td>Requirement within Use Case</td></tr><tr><td>WMS</td><td>Workshop Management System</td></tr><tr><td>2025</td><td>Year of document created</td></tr></table>		REQ	Requirement within Use Case	WMS	Workshop Management System	2025	Year of document created		
REQ	Requirement within Use Case									
WMS	Workshop Management System									
2025	Year of document created									

	1	Use Case Number
	01	Requirement Number
Traceability ID	ReqTrac-WMS-2025-101	
	ReqTrac	Requirement Traceability
	WMS	Workshop Management System
	2025	Year of document created
	1	Use Case Number
	01	Requirement Traceability Number
Sequence ID	SEQ-WMS-2025-101	
	SEQ	Sequence
	WMS	Workshop Management System
	2025	Year of document created
	1	Use Case Number
	01	Sequence Number

## 1.2 References

- [1] IEEE 830-1998, IEEE Recommended Practice for Software Requirements Specifications, IEEE Computer Society, 1998.
- [2] IEEE 1016-2009, IEEE Standard for Information Technology—Systems Design—Software Design Descriptions, IEEE Computer Society, 2009.
- [3] IEEE 1058-1998, IEEE Standard for Software Project Management Plans, IEEE Computer Society, 1998.
- [4] IEEE 1028-1997, IEEE Standard for Software Reviews, IEEE Computer Society, 1997.
- [5] ISO/IEC/IEEE 29148:2011, Systems and Software Engineering – Life Cycle Processes – Requirements Engineering, ISO/IEC/IEEE, 2011.
- [6] “Workshop Management System – POMEN.” Accessed: Mar. 30, 2025. [Online]. Available: <https://www.pomen.io/workshop-management-system/>

## 2.0 Formal Qualification Test Preparation

### 2.1 Testing Schedule

Test Case ID	Module	Start Date	End Date
TC-WMS-101	Manage Profile	02/06/2025	03/06/2025
TC-WMS-102		03/06/2025	03/06/2025
TC-WMS-103		04/06/2025	05/06/2025
TC-WMS-104		05/06/2025	05/06/2025
TC-WMS-105		06/06/2025	06/06/2025
TC-WMS-106		06/06/2025	07/06/2025
TC-WMS-107		07/06/2025	07/06/2025
TC-WMS-201	Manage Shop Inventory	04/06/2025	04/06/2025
TC-WMS-202		04/06/2025	05/06/2025
TC-WMS-203		05/06/2025	05/06/2025
TC-WMS-204		06/06/2025	06/06/2025
TC-WMS-205		06/06/2025	06/06/2025
TC-WMS-206		07/06/2025	07/06/2025
TC-WMS-301	Request Inventory	07/06/2025	07/06/2025
TC-WMS-302		08/06/2025	08/06/2025
TC-WMS-303		09/06/2025	09/06/2025
TC-WMS-401	Manage Foreman Schedule	04/06/2025	04/06/2025
TC-WMS-402		05/06/2025	05/06/2025
TC-WMS-403		05/06/2025	05/06/2025
TC-WMS-404		06/06/2025	06/06/2025
TC-WMS-405		06/06/2025	08/06/2025
TC-WMS-406		08/06/2025	09/06/2025
TC-WMS-407		10/06/2025	10/06/2025
TC-WMS-408		10/06/2025	10/06/2025
TC-WMS-501	Select Available Schedule	05/06/2025	05/06/2025
TC-WMS-502		05/06/2025	05/06/2025
TC-WMS-503		06/06/2025	06/06/2025
TC-WMS-504		06/06/2025	07/06/2025
TC-WMS-505		07/06/2025	07/06/2025
TC-WMS-506		08/06/2025	08/06/2025
TC-WMS-507		09/06/2025	10/06/2025
TC-WMS-508		10/06/2025	11/06/2025
TC-WMS-509		11/06/2025	11/06/2025
TC-WMS-601	Manage Payroll	04/06/2025	05/06/2025
TC-WMS-602		05/06/2025	06/06/2025
TC-WMS-603		07/06/2025	07/06/2025
TC-WMS-604		07/06/2025	08/06/2025
TC-WMS-605		08/06/2025	08/06/2025

TC-WMS-606		08/06/2025	10/06/2025
TC-WMS-607		10/06/2025	10/06/2025
TC-WMS-701	Manage Rating	06/06/2025	06/06/2025
TC-WMS-702		06/06/2025	06/06/2025
TC-WMS-703		07/06/2025	07/06/2025
TC-WMS-704		07/06/2025	08/06/2025
TC-WMS-705		09/06/2025	09/06/2025

## 2.2 Test Case of System

### 2.2.1 Manage Profile (Alya Syakirah – CB22141)

Test Case ID	Description
TC-WMS-101	This test is performed to verify that Workshop Owners and Foreman can successfully register with valid details.
TC-WMS-102	This test is performed to verify that users can log in using correct credentials.
TC-WMS-103	This test is performed to verify that users can view and edit their profile information.
TC-WMS-104	This test is performed to verify that profile updates are saved when valid data is entered.
TC-WMS-105	This test is performed to verify that errors are shown when required fields are invalid or empty.
TC-WMS-106	This test is performed to verify that clicking cancel discards unsaved profile changes.
TC-WMS-107	This test is performed to verify that the forgot password function sends a reset link to the user's email

## **2.2.2 Manage Shop Inventory (Varshini Jagarajan – CB22139)**

<b>Test Case ID</b>	<b>Description</b>
TC-WMS-201	This test is performed to verify that the owner can add a new inventory item by entering all required details.
TC-WMS-202	This test is performed to verify that the inventory items are listed correctly on the View Inventory page.
TC-WMS-203	This test is performed to verify that the owner can update details of an existing inventory item.
TC-WMS-204	This test is performed to verify that the owner can delete an inventory item from the system.
TC-WMS-205	This test is performed to verify that the supplier dropdown in the Add Inventory form retrieves supplier names from Firestore.
TC-WMS-206	This test is performed to verify that all required fields are validated and alerts are shown if any field is left empty.

## **2.2.3 Request Inventory (Varshini Jagarajan – CB22139)**

<b>Test Case ID</b>	<b>Description</b>
TC-WMS-301	This test is performed to verify that the owner can submit a request for an inventory item with valid input.
TC-WMS-302	This test is performed to verify that the owner can update the status of an existing inventory request.
TC-WMS-303	This test is performed to verify that the system alerts the user when required fields are left empty.

## 2.2.4 Manage Foreman Schedule (Amala Karthigayan – CB23116)

Test Case ID	Description
TC-WMS-401	Verify that the workshop owner can successfully add a new gig slot.
TC-WMS-402	Validate error message when attempting to add a slot with the same time/location/job (duplicate slot).
TC-WMS-403	Verify that the workshop owner can edit an existing gig slot.
TC-WMS-404	Confirm that fields like location and required foreman count are disabled in edit.
TC-WMS-405	Verify that the workshop owner can delete an existing gig slot.
TC-WMS-406	Check if deletion prompt appears before a gig slot is deleted.
TC-WMS-407	Verify that the workshop owner can view all applications submitted by foremen.
TC-WMS-408	Confirm that the owner can approve/reject a pending foreman application and see the status change.

## 2.2.5 Select Available Schedule (Amala Karthigayan – CB23116)

Test Case ID	Description
TC-WMS-501	Verify that a foreman can view available gigs for application.
TC-WMS-502	Validate that a foreman can apply for an available gig slot.
TC-WMS-503	Ensure the system checks if the slot is available before applying.
TC-WMS-504	Confirm that a successful application sends a confirmation message.
TC-WMS-505	Check that the foreman can view their applied gig slots.
TC-WMS-506	Validate that a foreman can cancel their application if more than 24 hours remain.
TC-WMS-507	Ensure that cancellation is not allowed within 24 hours of gig start.
TC-WMS-508	Validate the system notifies the owner upon cancellation by foreman.
TC-WMS-509	Ensures foreman receives notification upon approval/rejection of the gig slots applied.

## 2.2.6 Manage Payroll (Nabihah Fatiny – CB23101)

Test Case ID	Description
TC-WMS-601	This test is performed to verify that the Workshop Owner can view foremen with pending payroll records.
TC-WMS-602	This test ensures that a payment entry form is displayed when the Workshop Owner selects a foreman from the list.
TC-WMS-603	This test verifies that the Workshop Owner can successfully submit a valid payment with correct details.
TC-WMS-604	This test checks whether the system correctly displays an error message for invalid payment details such as incorrect account number.
TC-WMS-605	This test confirms that the Workshop Owner can modify the payment amount before final submission.
TC-WMS-606	This test ensures that the Workshop Owner can cancel a payment process before it is submitted.
TC-WMS-607	This test is performed to confirm that the Foreman can view their payroll records and payment status in the system.

## 2.2.7 Manage Rating (Nabihah Fatiny – CB23101)

Test Case ID	Description
TC-WMS-701	This test is performed to verify that the Customer can view a list of completed bookings for rating purposes.
TC-WMS-702	This test checks that the system displays a rating form when the Customer selects to submit a rating.
TC-WMS-703	This test is performed to ensure that the Customer can successfully submit a valid star rating and written comment.
TC-WMS-704	This test confirms that the Workshop Owner can access and view all ratings and comments submitted by customers.
TC-WMS-705	This test verifies that the Foreman can view all ratings and comments submitted by customers.

## **2.3 Hardware Preparation**

The mobile application for Workshop Inventory Management is designed to run on standard Android emulators. It does not require high-end hardware specifications but must operate on devices that support Android OS. The development and testing processes are carried out using computers equipped with Intel or AMD processors and at least 4GB of RAM. An active internet connection is necessary on both the development machine and mobile device to ensure synchronization with Firebase services. The application is optimized for general Android devices and can run smoothly on most modern smartphones.

***Hardware Requirement:***

<b>Category</b>	<b>Hardware</b>
Processor	Intel / AMD
Memory (RAM)	4GB or above
Internet Connectivity	Required
Supported Devices	Android Smartphones

## 2.4 Software Preparation

The mobile application for Workshop Inventory Management is created using the Flutter framework and connected to Firebase for its backend services. The app is created with Dart and developed using the Visual Studio Code editor. It is assessed and implemented on Android emulators through Android Studio. The system is built to operate on devices that use Android OS and needs an internet connection to synchronize with Firebase services.

### *Software Requirements:*

Category	Software
Programming Language	Dart
Framework	Flutter
Backend / Database	Firebase (Firestore)
Editor / IDE	Visual Studio Code 1.101
Emulator	Android Studio
Operating System	Windows 10 or above
Additional Tools	Firebase Console, Flutter SDK
Supported Devices	Android smartphones

### **3.0 Formal Qualification Test Description**

#### **3.1 Manage Profile (Alya Syakirah – CB22141)**

<b>Test Case ID</b>	TC-WMS-101
<b>Description</b>	Verify successful registration for Workshop Owners and Foreman.
<b>Precondition</b>	User not registered yet.
<b>Test Input</b>	<p>Click “Register”, fill in all required fields:</p> <ol style="list-style-type: none"><li>1. Foreman: Name, Phone Number, Position, Skills, Specialization, Experience</li><li>2. Workshop Owner: Name, Phone Number, Workshop Name, Workshop Address, Business Overview, Operating Hours</li></ol> <p>Then click “Submit”.</p>
<b>Expected Result</b>	User profile is saved in the Firestore and a success message is shown.
<b>Actual Result</b>	User registered successfully.

<b>Test Case ID</b>	TC-WMS-102
<b>Description</b>	Verify login with valid credentials.
<b>Precondition</b>	<ol style="list-style-type: none"><li>1. User account exists.</li><li>2. User is on the login page.</li></ol>
<b>Test Input</b>	Enter correct email and password, click “Login”.
<b>Expected Result</b>	User is redirected to the dashboard.
<b>Actual Result</b>	User logged in successfully.

<b>Test Case ID</b>	TC-WMS-103
<b>Description</b>	Verify viewing and editing profile information.
<b>Precondition</b>	User is logged in.
<b>Test Input</b>	Click the “Profile” section, then click “Edit” button.

<b>Expected Result</b>	Editable fields are displayed with current profile information.
<b>Actual Result</b>	Profile edit form shown with pre-filled data.

<b>Test Case ID</b>	TC-WMS-104
<b>Description</b>	Verify saving profile updates with valid data.
<b>Precondition</b>	User is logged in.
<b>Test Input</b>	Enter valid data in all required fields, click “Save Changes”.
<b>Expected Result</b>	Changes saved and success message displayed.
<b>Actual Result</b>	Profile updated successfully and changes reflected in the Firestore.

<b>Test Case ID</b>	TC-WMS-105
<b>Description</b>	Verify validation for empty or invalid input fields.
<b>Precondition</b>	User is logged in.
<b>Test Input</b>	Leave required fields empty or enter invalid email, click “Save Changes”.
<b>Expected Result</b>	System shows relevant error messages.
<b>Actual Result</b>	Errors displayed and profile not updated.

<b>Test Case ID</b>	TC-WMS-106
<b>Description</b>	Verify cancel button discards unsaved changes.
<b>Precondition</b>	User is editing their profile.
<b>Test Input</b>	Modify profile fields and click “Cancel”.
<b>Expected Result</b>	No changes are saved and the original profile data is shown.
<b>Actual Result</b>	Changes discarded.

<b>Test Case ID</b>	TC-WMS-107
<b>Description</b>	Verify forgot password function.
<b>Precondition</b>	User is on the login page.

<b>Test Input</b>	Click “Forgot Password”, enter registered email and click “Submit”.
<b>Expected Result</b>	Reset link is sent to the user’s email.
<b>Actual Result</b>	Password reset email received successfully.

### 3.2 Manage Shop Inventory (Varshini Jagarajan – CB22139)

<b>Test Case ID</b>	TC-WMS-201
<b>Description</b>	Verify adding a new inventory item
<b>Precondition</b>	<ol style="list-style-type: none"> <li>1. Owner logged in.</li> <li>2. Suppliers exist.</li> </ol>
<b>Test Input</b>	Click “Add Inventory”, fill all fields, click “Add Item”.
<b>Expected Result</b>	Item saved in Firestore and success message shown.
<b>Actual Result</b>	Item added successfully.

<b>Test Case ID</b>	TC-WMS-202
<b>Description</b>	View inventory list.
<b>Precondition</b>	1. Inventory items exist in Firestore.
<b>Test Input</b>	Click “View Inventory”.
<b>Expected Result</b>	Items listed with correct data.
<b>Actual Result</b>	Items displayed correctly.

<b>Test Case ID</b>	TC-WMS-203
<b>Description</b>	Edit inventory item.
<b>Precondition</b>	1. Existing inventory item.
<b>Test Input</b>	Tap item → Edit → change values → Save
<b>Expected Result</b>	Item updated and changes reflected in Firestore.
<b>Actual Result</b>	Item updated successfully.

<b>Test Case ID</b>	TC-WMS-204
<b>Description</b>	Delete inventory item.
<b>Precondition</b>	1. At least one item exists.
<b>Test Input</b>	Tap item → Delete → Confirm
<b>Expected Result</b>	Item removed from Firestore.
<b>Actual Result</b>	Item deleted.

<b>Test Case ID</b>	TC-WMS-205
<b>Description</b>	Supplier dropdown loads from Firestore.
<b>Precondition</b>	1. Supplier collection has records.
<b>Test Input</b>	Open Add Inventory → Click supplier dropdown
<b>Expected Result</b>	Supplier list retrieved and displayed.
<b>Actual Result</b>	Supplier list shown correctly.

<b>Test Case ID</b>	TC-WMS-206
<b>Description</b>	Validate required fields.
<b>Precondition</b>	1. Request form loaded.
<b>Test Input</b>	Leave quantity or supplier empty → click “Submit”
<b>Expected Result</b>	Alert: “Please fill all required fields.”
<b>Actual Result</b>	Alert shown.

### 3.3 Request Inventory (Varshini Jagarajan – CB22139)

<b>Test Case ID</b>	TC-WMS-301
<b>Description</b>	Submit inventory request.
<b>Precondition</b>	1. Owner selects low stock item.
<b>Test Input</b>	Enter quantity and select supplier → click “Submit”
<b>Expected Result</b>	Request saved to Firestore with status “Pending”.
<b>Actual Result</b>	Request submitted successfully.

<b>Test Case ID</b>	TC-WMS-302
<b>Description</b>	Change request status.
<b>Precondition</b>	1. At least one request exists.
<b>Test Input</b>	In Request Status page → choose status from dropdown
<b>Expected Result</b>	Status updates in Firestore.
<b>Actual Result</b>	Status updated.

<b>Test Case ID</b>	TC-WMS-303
<b>Description</b>	Validate required fields.
<b>Precondition</b>	1. Request form loaded.
<b>Test Input</b>	Leave quantity or supplier empty → click “Submit”
<b>Expected Result</b>	Alert: “Please fill all required fields.”
<b>Actual Result</b>	Alert shown.

### 3.4 Manage Foreman Schedule (Amala Karthigayan – CB23116)

#### TC-WMS-401 Add Gig Slot

Requirement: Workshop owner shall be able to add gig slots.

<b>Test Case ID</b>	TC-WMS-401
<b>Description</b>	Verify that the workshop owner can successfully add a new gig slot.
<b>Precondition</b>	1. Owner is logged in. 2. Owner is on <b>Add Gig Slot</b> page.
<b>Test Input</b>	Fill all required fields (date, time, location, job description, required foremen count, payment) → Click <<ADD SLOT>>.
<b>Expected Result</b>	New gig slot appears in the slot list; success message is shown.
<b>Actual Result</b>	New gig slot appears and a success message is shown.

### **TC-WMS-402 Duplicate-Slot Validation**

Requirement: Slots with same job description shall not exist at the same location, date and time.

<b>Test Case ID</b>	TC-WMS-402
<b>Description</b>	Validate error message when attempting to add a duplicate slot.
<b>Precondition</b>	<ol style="list-style-type: none"><li>1. A slot with date D, time T, location L, job J already exists.</li><li>2. Owner is on <b>Add Gig Slot</b> page.</li></ol>
<b>Test Input</b>	Enter <b>D, T, L, J</b> again → Click <<ADD SLOT>>.
<b>Expected Result</b>	System displays error “Duplicate slot ...” and prevents saving.
<b>Actual Result</b>	Error message is displayed stating that the slot already exists.

### **TC-WMS-403 Edit Gig Slot**

Requirement: Workshop owner shall be able to edit gig slots.

<b>Test Case ID</b>	TC-WMS-403
<b>Description</b>	Verify that the owner can modify editable fields of an existing gig slot.
<b>Precondition</b>	<ol style="list-style-type: none"><li>1. Owner is on <b>Edit Gig Slot</b> page for a valid slot.</li></ol>
<b>Test Input</b>	Change job name, description, time, date and payment details → Click <<SAVE CHANGES>>.
<b>Expected Result</b>	Slot is updated; list reflects new details.
<b>Actual Result</b>	Changes are made and listed successfully.

#### **TC-WMS-404 Read-Only Fields in Edit**

Requirement: Location and required foreman count cannot be modified once slot is created.

<b>Test Case ID</b>	TC-WMS-404
<b>Description</b>	Confirm location and foreman-count inputs are disabled in edit mode.
<b>Precondition</b>	1. Owner is on <b>Edit Gig Slot</b> page.
<b>Test Input</b>	Attempt to change location or required foremen.
<b>Expected Result</b>	Fields are read-only / disabled; no change possible.
<b>Actual Result</b>	The fields are disabled and can't make any changes.

#### **TC-WMS-405 Delete Gig Slot**

Requirement: Workshop owner shall be able to delete gig slots.

<b>Test Case ID</b>	TC-WMS-405
<b>Description</b>	Verify that owner can delete an existing gig slot.
<b>Precondition</b>	1. Owner views slot list.
<b>Test Input</b>	Select slot → Click <<Delete>> → Confirm prompt.
<b>Expected Result</b>	Slot is removed from list; success message shown.
<b>Actual Result</b>	Slot is successfully deleted and a message is shown.

### **TC-WMS-406 Deletion Prompt**

Requirement: Deletion requires confirmation.

<b>Test Case ID</b>	TC-WMS-406
<b>Description</b>	Check that a confirmation dialog appears before deletion.
<b>Precondition</b>	1. Same as TC-WMS-405.
<b>Test Input</b>	Click <<Yes>>.
<b>Expected Result</b>	Prompt “Are you sure to delete the gig slot?” appears; no deletion until confirmed.
<b>Actual Result</b>	Confirmation message is displayed before the deletion.

### **TC-WMS-407 View Applications**

Requirement: Workshop owner shall view all applications submitted by foremen.

<b>Test Case ID</b>	TC-WMS-407
<b>Description</b>	Verify that Manage Gig Application page lists every submitted application.
<b>Precondition</b>	1. At least one application exists.
<b>Test Input</b>	Navigate to <b>Manage Gig Applications</b> .
<b>Expected Result</b>	List shows each application with correct foreman details.
<b>Actual Result</b>	List of application with correct foreman details is shown.

### **TC-WMS-408 Approve/Reject Application**

Requirement: Workshop owner shall be able to approve or reject applications.

<b>Test Case ID</b>	TC-WMS-408
<b>Description</b>	Confirm approval/rejection updates status and notifies foreman.
<b>Precondition</b>	1. Owner opens <b>Manage Gig Application</b> for a “Pending” record.
<b>Test Input</b>	Click <<APPROVE>> or <<REJECT>> on submitted application.
<b>Expected Result</b>	Status changes to Approved/Rejected; foreman receives notification.
<b>Actual Result</b>	Foreman receives notification upon successful approval or rejection.

### **3.5 Select Available Schedule (Amala Karthigayan – CB23116)**

#### **TC-WMS-501 View Available Gigs**

Requirement: Foreman shall be able to view available gigs.

<b>Test Case ID</b>	TC-WMS-501
<b>Description</b>	Verify available gig list loads excluding past or full slots.
<b>Precondition</b>	1. Foreman is logged in.
<b>Test Input</b>	Navigate to <b>View Available Gigs</b> .
<b>Expected Result</b>	List displays open slots only.
<b>Actual Result</b>	Available gig slots are displayed successfully.

### **TC-WMS-502 Apply for Gig**

Requirement: Foreman shall be able to apply for an available gig slot.

<b>Test Case ID</b>	TC-WMS-502
<b>Description</b>	Validate successful application submission.
<b>Precondition</b>	1. Valid slot displayed in list.
<b>Test Input</b>	Select slot → Click <<APPLY>>.
<b>Expected Result</b>	Application status “Pending”; confirmation shown.
<b>Actual Result</b>	Foreman can successfully apply for the gig slot and a confirmation is shown.

### **TC-WMS-503 Availability Check**

Requirement: System must check slot availability before applying.

<b>Test Case ID</b>	TC-WMS-503
<b>Description</b>	Attempt to apply when slot is already full.
<b>Precondition</b>	1. Slot’s required foremen count reached.
<b>Test Input</b>	Click <<APPLY>> on full slot.
<b>Expected Result</b>	Error “Slot is full” displayed; no application created.
<b>Actual Result</b>	Error message is displayed and no application created.

### **TC-WMS-504 Application Confirmation**

Requirement: Foreman receives confirmation after applying.

<b>Test Case ID</b>	TC-WMS-504
<b>Description</b>	Ensure confirmation dialog appears post-apply.
<b>Precondition</b>	1. Same as TC-WMS-502.
<b>Test Input</b>	Apply for a slot.
<b>Expected Result</b>	Confirmation “Application submitted”.

<b>Actual Result</b>	Confirmation message is displayed along with a notification in the system.
----------------------	--

### **TC-WMS-505 View Applied Gigs**

Requirement: Foreman shall view their applied gig slots.

<b>Test Case ID</b>	TC-WMS-505
<b>Description</b>	Verify applied gigs appear in <b>AppliedGigList</b> .
<b>Precondition</b>	1. Foreman has at least one application.
<b>Test Input</b>	Navigate to <b>Applied Gigs</b> .
<b>Expected Result</b>	List displays status of each application.
<b>Actual Result</b>	The list of applied gigs is displayed with the application status (approved/rejected).

### **TC-WMS-506 Cancel Application (>24 h)**

Requirement: Foreman can cancel application if >24 h before gig.

<b>Test Case ID</b>	TC-WMS-506
<b>Description</b>	Validate cancellation when more than 24 hours remain.
<b>Precondition</b>	1. Gig start time is >24 h away.
<b>Test Input</b>	Select application → Click <<CANCEL>> → Confirm.
<b>Expected Result</b>	Status changes to Cancelled; owner notified.
<b>Actual Result</b>	Cancellation is successful and the owner receives a notification.

### **TC-WMS-507 Cancellation Not Allowed (<24 h)**

Requirement: Cancellation not allowed within 24 h of gig start.

<b>Test Case ID</b>	TC-WMS-507
<b>Description</b>	Attempt cancellation with <24 h remaining.
<b>Precondition</b>	1. Gig starts in <24 h.
<b>Test Input</b>	Select application → Click <<CANCEL>>
<b>Expected Result</b>	Error “Cannot cancel within 24 h” shown; no status change.
<b>Actual Result</b>	Error message is displayed.

### **TC-WMS-508 Owner Notification on Cancellation**

Requirement: Owner notified when foreman cancels

<b>Test Case ID</b>	TC-WMS-508
<b>Description</b>	Verify owner receives notification after cancellation.
<b>Precondition</b>	1. Cancellation confirmed (TC-WMS-506).
<b>Test Input</b>	Observe owner notification feed.
<b>Expected Result</b>	Notification “Foreman X cancelled slot Y”.
<b>Actual Result</b>	Notification is received by the respective owners.

### **TC-WMS-509 Foreman Notification on Approval/Rejection**

Requirement: Foreman notified upon approval/rejection.

<b>Test Case ID</b>	TC-WMS-509
<b>Description</b>	Confirm foreman receives notification when owner approves/rejects application.
<b>Precondition</b>	1. Owner makes decision (TC-WMS-408).
<b>Test Input</b>	Observe foreman notification center.

<b>Expected Result</b>	Notification “Application Approved” or “Application Rejected”.
<b>Actual Result</b>	Foreman successfully receives notification upon the decision made by the owners and the application status changes according to it.

### 3.6 Manage Payroll (Nabihah Fatiny – CB23101)

Requirement: Workshop Owner can view pending payroll for each foreman.

<b>Test Case ID</b>	TC-WMS-601
<b>Description</b>	Verify Workshop Owner can view foremen with pending payments
<b>Precondition</b>	Workshop Owner is logged in. Pending payments exist.
<b>Test Input</b>	<ol style="list-style-type: none"> <li>1. Click “Manage Payroll” from the dashboard</li> <li>2. System loads a list of foremen with pending payments.</li> </ol>
<b>Expected Result</b>	List of foremen with pending payments is displayed
<b>Actual Result</b>	List of pending foremen appears successfully on the screen.

Requirement: System displays payment entry form when a foreman is selected.

<b>Test Case ID</b>	TC-WMS-602
<b>Description</b>	Verify payment form displays after selecting a foreman
<b>Precondition</b>	Pending payment exists for the selected foreman
<b>Test Input</b>	<ol style="list-style-type: none"> <li>1. From the “Manage Payroll” list, click on a foreman with a pending payment.</li> <li>2. System displays the payment entry form with required fields: Bank, Name, Account Number, Amount, Date Reference</li> </ol>

<b>Expected Result</b>	Payment form with fields appears (Bank, Name, Account No., Amount, Date, Reference)
<b>Actual Result</b>	Payment entry form is shown with all fields visible and ready for input.

Requirement: Workshop Owner can make payment to foreman using valid input.

<b>Test Case ID</b>	TC-WMS-603
<b>Description</b>	Submit a valid payment successfully
<b>Precondition</b>	Form filled with valid details
<b>Test Input</b>	<ol style="list-style-type: none"> <li>1. Click on a foreman, fill in all required fields in the payment form:</li> <li>2. click “Submit Payment”</li> </ol>
<b>Expected Result</b>	Payment processed, success message displayed, status = “Successful”
<b>Actual Result</b>	Payment submission successful and confirmation message appears with status marked as “Successful”.

Requirement: System validates payment input and shows error for incorrect account details.

<b>Test Case ID</b>	TC-WMS-604
<b>Description</b>	Handle invalid payment details
<b>Precondition</b>	Payment form is opened
<b>Test Input</b>	<ol style="list-style-type: none"> <li>1. Click on a foreman, fill in the form with invalid account number.</li> <li>2. Click Submit button</li> </ol>
<b>Expected Result</b>	Error message: “Invalid payment details. Please check and try again.”
<b>Actual Result</b>	System displays validation error and prevents submission due to incorrect account details.

Requirement: Workshop Owner can update payment amount before submitting.

<b>Test Case ID</b>	TC-WMS-605
<b>Description</b>	Modify payment amount before submission
<b>Precondition</b>	Payment form is displayed
<b>Test Input</b>	<ol style="list-style-type: none"> <li>1. Click on a foreman to open the form.</li> <li>2. Fill in all fields with a default amount of RM1000.00.</li> <li>3. Click “Update”, change the amount to RM1200.00.</li> <li>4. Then click “Save Changes”.</li> </ol>
<b>Expected Result</b>	Updated amount saved and shown in form
<b>Actual Result</b>	Modified payment amount is updated and displayed in the form successfully.

Requirement: Workshop Owner can cancel a payment before it's finalized.

<b>Test Case ID</b>	TC-WMS-606
<b>Description</b>	Cancel payment after starting submission
<b>Precondition</b>	Form is filled and submission started
<b>Test Input</b>	Click “Cancel” after clicking “Submit Payment”
<b>Expected Result</b>	Payment process is cancelled. no record is submitted
<b>Actual Result</b>	Payment action is aborted and no data is recorded in the system.

Requirement: Foreman can view his payroll history and payment status.

<b>Test Case ID</b>	TC-WMS-607
<b>Description</b>	Foreman views their payroll records
<b>Precondition</b>	Foreman is logged in. payroll records exist
<b>Test Input</b>	<ol style="list-style-type: none"> <li>1. Foreman logs into the system.</li> <li>2. Click “Manage Payroll” from the navigation menu.</li> </ol>

	3. System displays personal payroll records and statuses.
<b>Expected Result</b>	Records and payment status are shown
<b>Actual Result</b>	Payroll history and statuses are successfully shown to the foreman.

### 3.7 Manage Rating (Nabihah Fatiny – CB23101)

Requirement: Customers can view their completed bookings to submit ratings

<b>Test Case ID</b>	TC-WMS-701
<b>Description</b>	Verify that Customer can view completed bookings
<b>Precondition</b>	Booking is marked as completed and rating feature is active
<b>Test Input</b>	Customer navigates to “Manage Rating” section
<b>Expected Result</b>	A list of completed bookings is presented to the customer
<b>Actual Result</b>	Completed bookings are displayed clearly for rating selection

Requirement: Customers can access a rating form after selecting a completed booking

<b>Test Case ID</b>	TC-WMS-702
<b>Description</b>	Verify rating form is shown when submitting a rating
<b>Precondition</b>	Completed booking exists
<b>Test Input</b>	Customer clicks “Submit Your Rating” button
<b>Expected Result</b>	Rating form with stars and feedback field is shown
<b>Actual Result</b>	Star rating interface and feedback text area appear successfully

Requirement: Customers can submit a star rating and written feedback after a service.

<b>Test Case ID</b>	TC-WMS-703
<b>Description</b>	Submit a valid star rating and feedback comment
<b>Precondition</b>	Rating form is visible after click “Submit Your Rating” button
<b>Test Input</b>	Customer selects 5 stars and writes a comment, then clicks submit
<b>Expected Result</b>	The system saves the rating and displays it on the Viewr Rating Page
<b>Actual Result</b>	Submitted rating and comment are recorded and shown properly

Requirement: Workshop Owners can view all customer-submitted ratings and comments.

<b>Test Case ID</b>	TC-WMS-704
<b>Description</b>	View submitted ratings as Workshop Owner
<b>Precondition</b>	Customer ratings are available
<b>Test Input</b>	Workshop Owner navigate to “Manage Rating” section
<b>Expected Result</b>	All customer feedback and ratings are listed on the screen
<b>Actual Result</b>	A compiled list of customer feedback and ratings

Requirement: Foremen can view all customer-submitted ratings and comments.

<b>Test Case ID</b>	TC-WMS-705
<b>Description</b>	View submitted ratings as Foreman
<b>Precondition</b>	Ratings exist in the system
<b>Test Input</b>	Foreman navigates to the “Manage Rating” page
<b>Expected Result</b>	Ratings given by customers are viewable in read-only mode
<b>Actual Result</b>	Foreman can successfully view ratings and feedbacks

#### **4.0 Acronym and Abbreviation**

<b>Acronyms / Abbreviation</b>	<b>Description</b>
STD	Software Test Description
WMS	Workshop Management System
TC	Test Case
REQ	Requirement