

# **Master Test Plan**

White Rose Interiors WhatsApp Chatbot

## **Group 1**

University of the West Indies

December 3, 2025  
Version 2.5 (Landscape)

<b>Item</b>	<b>Details</b>
<b>Project Name</b>	White Rose Interiors Chatbot
<b>Document Type</b>	Master Test Plan (MTP)
<b>Version</b>	2.5 (Landscape Orientation)
<b>Status</b>	<b>DRAFT</b>

# Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
1.1	Objectives . . . . .	3
<b>2</b>	<b>Test Environment</b>	<b>3</b>
<b>3</b>	<b>Entry &amp; Exit Criteria</b>	<b>3</b>
3.1	Entry . . . . .	3
3.2	Exit . . . . .	4
<b>4</b>	<b>Test Case Execution Log</b>	<b>5</b>
4.1	Requirement 1: User Interface & Flows . . . . .	5
4.2	Requirement 2: Price Calculator . . . . .	5
4.3	Requirement 3: Invoicing . . . . .	6
4.4	Requirement 4: Admin Dashboard . . . . .	6
4.5	Requirement 5: Status Updates . . . . .	6
4.6	Requirement 6: Automated Reminders . . . . .	7
4.7	Requirement 7: Manage Orders (Self-Service) . . . . .	7
4.8	Data Integrity & Performance . . . . .	8
<b>5</b>	<b>Requirements Traceability Matrix</b>	<b>9</b>
<b>6</b>	<b>Defect Report Template</b>	<b>10</b>
<b>7</b>	<b>Approvals</b>	<b>11</b>

# 1 Introduction

This Master Test Plan (MTP) outlines strategy, resources, and procedures for testing the White Rose Interiors Chatbot. The system is a Node.js app using `whatsapp-web.js` and local JSON storage.

## 1.1 Objectives

- Verify all functional requirements (UI, pricing, invoicing, dashboard, status, reminders, manage orders).
- Validate file-based storage integrity across updates (`orders.json`, `cleaning_supply.json`, `reminders.json`).
- Ensure chatbot/dash respond within a few seconds under normal load and recover from invalid inputs.

# 2 Test Environment

Component	Specification
<b>Hardware</b>	Windows 10/11 or Server 2019+
<b>Runtime</b>	Node.js v18+
<b>Storage</b>	Local File System ( <code>orders.json</code> , <code>cleaning_supply.json</code> , <code>reminders.json</code> )
<b>Mobile</b>	1 Android + 1 iOS for WhatsApp tests
<b>Browser</b>	Chrome/Edge for Admin Dashboard

Table 1: Test Environment Specifications

# 3 Entry & Exit Criteria

## 3.1 Entry

- `server.js` starts without errors; WhatsApp client QR scanned and “Ready”.
- Baseline data present in `orders.json`, `cleaning_supply.json`.
- Admin token obtainable via `/api/login`.

### **3.2 Exit**

- 100% High-priority tests passed; no open High/Critical defects.
- PDFs generate on host FS; reminder logs written to `reminders.json`.

## 4 Test Case Execution Log

### 4.1 Requirement 1: User Interface & Flows

ID	Scenario	Steps	Expected	Stat	Actual Result	Comments
UI-01	Main Menu	1) Send "Hi".	Shows: Goods/Services, Manage Orders, Get Order Status, Contact Support, FAQ.			
UI-02	Invalid Input	1) From menu, send "99".	Bot prompts to pick a listed option.			
UI-03	Cleaning Services	1) Select Goods/Services □ Cleaning Services. 2) Send "Need sofa cleaning".	Bot replies that Cleaning Services are currently unavailable, notes the request, and state resets.			
UI-04	Cleaning Supplies	1) Select Goods/Services □ Cleaning Supplies. 2) Choose a category from cleaning_supply.json poll.	Bot prompts "Please leave a message describing what you need," then confirms receipt.			

### 4.2 Requirement 2: Price Calculator

ID	Scenario	Steps	Expected	Stat	Actual Result	Comments
PC-01	Blind Quote	1) Select product. 2) Enter valid width/height.	Price matches grid + install; PDF estimate sent.			
PC-02	Invalid Dim	1) Enter "-5".	Bot rejects, asks for valid number.			
PC-03	Email Estimate	1) Complete quote. 2) Choose "Email me".	Email prompted, PDF emailed (or graceful failure message).			

### 4.3 Requirement 3: Invoicing

ID	Scenario	Steps	Expected	Stat	Actual Result	Comments
INV-01	Confirm Order	1) From estimate, choose "Confirm Order".	New order saved in orders.json with correct status (e.g., PENDING DELIVERY).			
INV-02	Email Failure	1) Request email when SMTP unavailable.	Bot says "Could not send email...", logs error.			

### 4.4 Requirement 4: Admin Dashboard

ID	Scenario	Steps	Expected	Stat	Actual Result	Comments
AD-01	Auth Negative (All Routes)	1) Access /api/orders, /api/report, /api/status without token.	All protected endpoints return 401 Unauthorized.			
AD-02	Update Order	1) Change status to READY.	Persisted to orders.json; notify ready sends WA msg.			
AD-03	Reminder UI	1) Edit reminder time in table.	Picker stays open; saved time matches selection.			
AD-04	Filters/Report	1) Apply text/status/date filters. 2) Download report.	Filters apply; CSV downloads successfully.			

### 4.5 Requirement 5: Status Updates

ID	Scenario	Steps	Expected	Stat	Actual Result	Comments
ST-01	Status Lookup	1) Select "Get Order Status".	Returns orders for caller's phone only.			

ID	Scenario	Steps	Expected	Stat	Actual Result	Comments
ST-02	Status Change	1) Admin updates status to "In Progress" or "Ready".	Customer receives WhatsApp notification; status persists in orders.json.			
ST-03	Job Completion	1) Admin updates status to "Delivered" or "Picked Up".	Customer receives the status-change notification; status persists.			

#### 4.6 Requirement 6: Automated Reminders

ID	Scenario	Steps	Expected	Stat	Actual Result	Comments
REM-01	Send Reminder	1) Set reminderTime due soon. 2) Wait for scheduler.	WA reminder sent; reminderSent=true; reminders.json log written.			
REM-02	Reset Flag on Edit	1) Reminder already sent (true). 2) Admin updates time.	reminderSent resets to false in JSON to allow re-send.			

#### 4.7 Requirement 7: Manage Orders (Self-Service)

ID	Scenario	Steps	Expected	Stat	Actual Result	Comments
MG-01	Retrieve Booking	1) Select "Manage Orders".	Bot lists caller's upcoming orders (max 10).			
MG-02	Reschedule	1) Manage Orders □ select order □ Reschedule.	New date/time saved to orders.json; dashboard reflects update.			
MG-03	Cancel Booking	1) Manage Orders □ Cancel □ confirm.	Order removed/updated; bot confirms; dashboard reflects change.			

<b>ID</b>	<b>Scenario</b>	<b>Steps</b>	<b>Expected</b>	<b>Stat</b>	<b>Actual Result</b>	<b>Comments</b>
MG-04	Invalid Selection	1) In Manage Orders, choose invalid option.	Bot prompts to select a valid order/action.			

## 4.8 Data Integrity & Performance

<b>ID</b>	<b>Scenario</b>	<b>Steps</b>	<b>Expected</b>	<b>Stat</b>	<b>Actual Result</b>	<b>Comments</b>
INT-01	JSON Integrity	1) Corrupt cleaning_supply.json.	Bot handles error/logs; no crash (graceful fallback).			
INT-02	Order JSON Corrupt	1) Manually corrupt orders.json (bad syntax). 2) Restart.	Error logged; service does not crash, but data load fails gracefully (no backup expected).			
PERF-01	Response SLA	1) Send 5 rapid messages.	Replies within a few seconds each under normal load.			

## 5 Requirements Traceability Matrix

Requirement ID	Associated Test Cases
<b>Req 1: User Interface</b>	UI-01, UI-02, UI-03, UI-04
1.1 Display service options	UI-01
1.2 Select options via number/keyword	UI-01, UI-04
1.3 Invalid input detection	UI-02, MG-04
1.4 Correct next-step routing	UI-03, UI-04
<b>Req 2: Price Calculator</b>	PC-01, PC-02, PC-03
2.1 Generate estimate	PC-01
2.2 Send quotation via WA/email	PC-01, PC-03
<b>Req 3: Invoice Generation</b>	INV-01, INV-02
3.1 Generate invoice	INV-01
3.2 Store invoice/order details	INV-01
3.3 Email invoice	INV-01, INV-02
<b>Req 4: Admin Dashboard</b>	AD-01, AD-02, AD-03, AD-04
4.1 Generate report	AD-04
4.2 Filter dashboard data	AD-04
4.3 Display totals, unresolved tickets	AD-04
<b>Req 5: Status Updates</b>	ST-01, ST-02, ST-03
5.1 Query status	ST-01
5.2 Retrieve job status	ST-01
5.3 Proactive updates	ST-02
5.4 Final message after completion	ST-03

<b>Req 6: Automated Reminders</b>	REM-01, REM-02
6.1 Scan upcoming appointments	REM-01
6.2 Send reminder messages	REM-01
6.3 Confirm/reschedule options	REM-01, MG-02
6.4 Log reminder sent	REM-01, REM-02
<b>Req 7: Manage Orders</b>	MG-01, MG-02, MG-03, MG-04
7.1 Retrieve upcoming appointments	MG-01
7.2 Authenticate by phone number	MG-01
7.3 Present reschedule/cancel options	MG-02, MG-03
7.4 Reschedule flow	MG-02
7.5 Cancel flow	MG-03
<b>Non-functional: Performance</b>	PERF-01
<b>Non-functional: Data Integrity</b>	INT-01, INT-02

## 6 Defect Report Template

Defect ID	Test ID	Sev	Description of Failure

## 7 Approvals

**QA Lead:**

---

(Signature / Date)

**Developer Lead:**

---

(Patrick Marsden)

**Client Representative:**

---

(White Rose Interiors)