Project Management Plan (PMP)

# 1. Scope

Project Name: Car Purchasing Web Application  
  
Objective: To build a platform that allows sellers to list and market their cars, and buyers to search, view, and reserve cars based on various criteria.  
  
**In Scope:**

* Requirement analysis
* High level design
* Low level design
* development
* Testing(validations)

**Out of Scope:**

* Unit testing
* Integration testing
* hosting
* Maintenance

# 2. Constraints & Assumptions

**Constraints:**

* WebApp PC based
* Unique user ID
* Admin features needed (create user, delete )

**Assumptions:**

* WebApp will work only in google chrome browser

# 3. Roles & Responsibilities (R&R)

|  |  |  |
| --- | --- | --- |
| Name | Role | Responsibilities |
| Ebrahem Mostafa | Project Manager | Plan, monitor, and control the project. Facilitate communication, manage risks, and ensure timely delivery. |
| Omar Abdou | Developer | Implement backend and frontend features. Ensure code quality and integration. |
| Arsany Mounir | Developer | Work on frontend/backend tasks. Collaborate with tester and designer for functionality. |
| Abdelaziz Elsisi | Tester | Prepare test cases, perform testing, report bugs, and ensure the product meets requirements. |
| Anas Mohamed | Tester | Support test planning and execution. Perform regression and sprint testing. |
| Ehab Ahmed | Designer | Create UI/UX designs, wireframes, and ensure visual consistency of the website. |

# 4.Configuration Management (CM)

Version Control System: Git  
Repository Platform: GitHub  
Repository Link: https://github.com/ebrahem1812/Car-Purchasing  
  
Branching Strategy:  
- Main Branch: Stable version, reflects the latest approved version.  
- Development Branch: Ongoing work, feature development, and updates.  
  
Configuration Items Tracked:  
- Source code  
- Design files  
- PMP documents  
- Requirement specs  
- Testing artifacts  
- Review documents  
  
Change Tracking:  
- Changes are tracked using commits with clear messages.  
- Team members contribute via pull requests or direct commits (based on internal agreements).  
- History of changes is maintained and reviewed via Git logs.

# 5. Phases (Sprints Breakdown)

|  |  |  |
| --- | --- | --- |
| Sprint | Week | Focus |
| Sprint 1 | Week 1 | Initial planning: SIQ, CRS, PMP |
| Sprint 2 | Week 2 | SRS and project timeline definition |
| Sprint 3 | Week 3 | UI/UX Design: Wireframes, mockups, and user flow creation |
| Sprint 4 | Week 4 | Frontend & Backend development (Phase 1): User registration, login, car listing |
| Sprint 5 | Week 5 | Development (Phase 2): Search functionality, reservation system, admin features |
| Sprint 6 | Week 6 | Testing: Functional, usability, and regression testing. Bug fixing. Final delivery |

# 6. Task Management (Jira)

The team uses **Jira** to manage and track all project tasks. Each work item—such as creating documents, developing features, or testing—is logged as a separate issue.

Tasks follow a structured flow: **To Do → In Progress → Done**.

Jira also supports:

* Sprint planning and tracking
* Assigning responsibilities
* Reviewing and baselining deliverables

This ensures transparency, accountability, and smooth progress throughout the project.

# 7. Change Requests (CR)

Change Requests Process:  
- All change requests must be submitted through a formal CR Form or via documented email/meeting notes.  
- The Project Manager (Ebrahem Mostafa) reviews the request and assesses its impact on scope, cost, and schedule.  
- Based on the analysis, the change is either approved or rejected.  
- Approved changes are added to the backlog for future sprints.

# 8. Risk Management

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Description | probability | impact | Score | Prevention Technique |
| Developer Absence | 1 | 4 | 4 | 2 backup team members by [19/4/2024]. |
| Security vulnerabilities | 3 | 5 | 15 | Transfer to security department |

# 9. Reviews

**Internal Team Reviews:**  
- Every 2 days: Team members meet on Discord for short sync meetings to review progress, align tasks, and raise blockers.  
- Daily follow-ups via WhatsApp group to ensure continuous communication and quick updates.  
  
**Sprint Review Meetings:** - End of every sprint: A formal review meeting will be held to  
 Demonstrate sprint deliverables  
 Get team feedback  
 Prepare for the next sprint  
  
**Customer Reviews:**- Sprint review outcomes will be shared with the customer for feedback.  
- All feedback will be documented and considered for the next sprint or raised as CRs.

# 10. Communication Plan (Meetings & Tools)

|  |  |  |  |
| --- | --- | --- | --- |
| Type | Tool / Platform | Frequency | Purpose |
| Daily Follow-Up | WhatsApp | Daily | Quick status updates, blockers, coordination |
| Team Sync Meeting | Discord | Every 2 days | Discuss progress, align tasks, clarify blockers |
| Sprint Review Meeting | Discord / In-person | End of each sprint | Review deliverables, gather feedback, plan next sprint |
| Customer Communication | Email | Weekly (or as needed) | Share progress, review documents, collect feedback |
| Documentation Sharing | GitHub | As needed | Share and collaborate on project documents |

# 11. Quality Management

**Goal:**  
Ensure the Car Purchasing Web Application meets all functional, performance, and user expectations.

**Main Objectives:**

* Core features (login, search, reservation) must work smoothly.
* The app should be user-friendly, responsive, and secure.
* User data must be protected with proper authentication.

**Activities:**

* **Quality Assurance (QA):** Code reviews, design checks, and test planning.
* **Quality Control (QC):** Functional testing, and fixing bugs.

**Team Roles:**

* PM manages quality efforts.
* Testers test the app.
* Developers fix issues.
* Users give feedback on UAT.

**Improvement:**

* Collect feedback, review results, and apply lessons learned.

# 11. Document Naming Conventions

To maintain consistency, traceability, and clarity across all project artifacts, the following naming conventions will be used for project documents:

|  |  |  |
| --- | --- | --- |
| Document Type | Naming Convention | Example |
| Project Management Plan | PMP\_<ProjectName>\_<Version>.docx | PMP\_CarPurchasing\_1.0.docx |
| Requirements Document | REQ\_<ModuleName>\_<Version>.docx | REQ\_CRS\_1.0.docx |
| Software Design Document | SDD\_<ModuleName>\_<Version>.docx | SDD\_UserLogin\_1.0.docx |
| Test Plan | TestPlan\_<Sprint/Module>.docx | TestPlan\_Sprint4\_1.0.docx |
| Test Case Document | TestCases\_<Function/Module>.xlsx | TestCases\_Reservation\_1.0.xlsx |
| Review Documents | Review\_<DocType>\_<Module>.docx | Review\_CR\_Reservation.docx |
| Change Request | CR\_<ChangeTitle>\_<Date>.docx | CR\_AddAdminPanel\_2025-04-10.docx |
| Meeting Minutes | Minutes\_<Topic>\_<Date>.docx | Minutes\_Sprint1Kickoff\_2025-04-01.docx |
| UI Design Files | UI\_<ScreenName>\_<Version>.fig | UI\_Homepage\_1.0.fig |

Versioning should follow the format, Major.Minor (e.g., 1.0, 1.1), and documents should be updated and baselined after every approved review cycle.