Product Planning Document

for

Cruise Connect – A Car Rental Website

The Team:

N Digvijay (PES2UG21CS310)
Nihal T M (PES2UG21CS333)
Nishanth D'Mello (PES2UG21CS343)
Rohith Hunashikatti (PES2UG21CS920)

PES University
22nd October, 2023

1. Lifecycle:

Agile Methodology with Rapid Development Cycle

Planning: Gathering requirements, defining project scope, and creating a brief project plan.
Design: Developing UI wireframes and outlining the system's architecture.
Development: Building essential functionalities (User Authentication, Admin Dashboard, Car Listing, and Basic Booking System).
Deployment: Preparing the platform for testing on a hosting environment.
Maintenance: Addressing critical issues and enhancing features based on feedback.

This lifecycle is chosen because it is a well-established and proven approach to software development. It allows for careful planning, design, and testing, which is essential for a project like building a car rental website.

2. Tools:

Design tool: draw.io
Version control: Git for code versioning, GitHub for collaboration.
Development tool: React.js, Node.js, MongoDB for frontend, backend, and database.
Bug tracking: Jira Software Management for issue management.
Testing tool: Jest, React Testing Library for frontend testing, Supertest for backend.

These tools are chosen because they are widely used and well-supported. They are also well-suited for the specific tasks involved in developing a website.

3. Deliverables:

Build Components:
User Authentication Module
Admin Dashboard with CRUD
Basic Car Listing & Filtering
Booking System (without payment integration)
Other Deliverables: Project Plan, Design Wireframes, Basic Test Cases, Initial Documentation.

4. WBS (Work Breakdown Structure):

Planning:
Requirements Gathering
Project Scope Definition
Brief Project Plan Creation
Design:
UI Wireframing
High-level System Architecture Design

Development:
User Authentication Implementation
Admin Dashboard Implementation
Car Listing & Filtering Development
Basic Booking System Implementation
Deployment:
Preparation for Testing Environment Setup
Maintenance:
Ongoing Bug Fixes and Feature Enhancements

5. Effort Estimation:

Planning: 1 week
Design: 1 week
Development: 2-3 weeks
Deployment: 1 week
Maintenance: Ongoing

6. Gantt Chart (Sample):

Task	Start	End
Planning	2023-11-01	2023-11-03
Design	2023-11-04	2023-11-05
Development	2023-11-06	2023-11-08
Deployment	2023-11-09	2023-11-10
Maintenance	2023-11-11	Ongoing

Justification:

The chosen approach aligns with the project's short timeframe, focusing on rapid development while ensuring essential functionalities.
Tools selected are popular, user-friendly, and suitable for quick development cycles.
The deliverables concentrate on core components necessary for the MVP within the timeframe.
Effort estimation and timelines are compact to fit within the sprint without compromising quality.
This streamlined plan aligns with the project's constraints, ensuring efficient development and delivery of a functional prototype within the one-week sprint. Adjustments can be made based on resource availability and project dynamics.