**Configuration Management Plan**

**1**. **Introduction**

The Configuration Management Plan (CMP) outlines the procedures, tools, and responsibilities for managing the configuration of Car Purchasing Web Application.

**2. Purpose**

The purpose of this CMP is to:

* Define the configuration management process for Car Purchasing Web Application.
* Establish roles and responsibilities for configuration management activities.
* Identify configuration items and their attributes.
* Describe the tools and techniques used for configuration management.

**3. Configuration Management Process**

**3.1 Configuration Identification**

* Identify configuration items (CIs) including software and     documentation.
* Establish baselines for each CI at key milestones.

**3.2 Configuration Control**

* Document change requests and their impact on project baselines.
* Establish a configuration control board (CCB) responsible for reviewing and approving changes.

**4. Roles and Responsibilities**

* Project Manager: Overall responsibility for configuration management within the project.
* Development Team: Responsible for adhering to configuration management procedures,
* Configuration Control Board (CCB): Reviews and approves proposed changes to configuration items.

**5. Tools and Techniques**

* Version Control System (VCS): Git will be used as the primary version control tool for managing source code and documentation.
* Document Management System: Google docs will be used for storing and managing project documentation.

**6. Documentation**

* Configuration Management Plan
* Configuration Item List

**7. Branching strategy**

* **Main Branch (Master)** The primary branch always reflects the production-ready state, changes made through pull/merge requests.
* **Development Branch (Develop):** All ongoing development work merged here; developers create feature branches from here.
* **Feature Branches:** Dedicated to specific features or tasks, changes merged back into the development branch.
* **Release Branches:** Created for stabilizing and preparing code for deployment, merged into the main and development branches when ready.

**8. Baseline Strategy:**

- Baselines are established at key milestones in the development process.

- Baselines ensure that configurations at specific points in time are identifiable and retrievable.

- Baselines are used as reference points for comparison and to track changes over time.