Car Rental Project

# Project Overview

This project aims to build a system for running a Car rental operation.

The car rental operation will function as the whole system for the entire brick and mortar operation. The Car Rental System will be designed to provide a view for a cashier, a view for a customer and a view for a manager. The car rental system will provide a few more back end features, the ability to breakdown a car, a gps tracking system, and a system to track car availability among other things.

# Team Organization

We will be using agile development techniques, rotating through scrum masters as the project progresses. The goal is to keep the team organization as simple as possible. Tasks will be assigned during stand up and everyone is given the tasks based on story points. A central idea of our team organization is that we will be all equal team members with the ability to pick up any task. Each member should be equal and able to take any task that needs done should it be required.

# Software Development Process

The development will be broken up into five phases. Each phase will be a little like a Sprint in an Agile method and a little like an iteration in a Spiral process. Specifically, each phase will be like a Sprint, in that work to be done will be organized into small tasks, placed into a “backlog”, and prioritized. Then, using on time-box scheduling, the team will decide which tasks the phase (Sprint) will address. The team will use a Scrum Board to keep track of tasks in the backlog, those that will be part of the current Sprint, those in progress, and those that are done.

Each phase will also be a little like an iteration in a Spiral process, in that each phase will include some risk analysis and that any development activity (requirements capture, analysis, design, implementation, etc.) can be done during any phase. Early phases will focus on understanding (requirements capture and analysis) and subsequent phases will focus on design and implementation. Each phase will include a retrospective.

|  |  |
| --- | --- |
| **Phase** | **Iteration** |
| 1. | Phase 1 - Requirements Capture |
| 2. | Phase 2 - Analysis, Architectural, UI, and DB Design |
| 3 | Phase 3 - Implementation, and Unit Testing |
| 4 | Phase 4 - More Implementation and Testing |

We will use Unified Modeling Language (UML) to document user goals, structural concepts, component interactions, and behaviors.

# Communication policies, procedures, and tools

Communication will be done via discord. Scrum meetings will be either over calls or in person in class. Other than that we will have Jira to organize tasks and use git for collaboration. Git procedure will be to make changes on dev and then merge dev into main for milestones. Main should always be operable, and pushes should be treated as a release.

# Configuration Management

See the README.md in the Git repository.

# Risk Management

Risk management will be conducted on our side by only releasing full, functional versions and keeping main fully operable. Along with that a typical risk for a project like this is scope creep, and we are keeping a list of requirements following the MCSW requirements outline.