Dan's Rent a Jeep

Project Overview

This project aims to build a system which allows five different users to access a rent-a-car system. The five users will each have different permissions in the system which will allow them to accomplish their respective goals.

The system will allow the owner to pay his employees, a manager to manage the fleet of vehicles available to the customer, a desk clerk to manage reservations, a vehicle recovery specialist to locate and return vehicles, and a customer to view and reserve an available vehicle from the fleet.

Team Organization

Project Manager - Clark Farnsworth
Team Members - Chase Cheney, Stephen Engst, Eric Wu

The project manager will delegate tasks to team members from our list of requirements. The project manager will also get a chance to pass their role to another team member for every milestone if they choose.

All members can create and merge pull GitHub repository pull requests.

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

We will communicate using Discord to plan meetings and to discuss any related information. We may also meet in the library or on Discord on Mondays after 4 PM for stand up meetings and group work when not meeting in class.

Google Drive and its associated tools will be used to collaborate on documents.

GitHub will be used for final submissions to the customer and version control.

Jira will be used for managing our roadmap and tasks/issues.

Configuration Management

See the README.md in the Git repository.

Risk Analysis

- Incomplete User Interface
    - Likelihood: Unlikely
    - Severity: Significant
    - Consequences: Inability of users to access application
    - Work-around: Provide backup interface
- Database Loss
    - Likelihood: Unlikely
    - Severity: Major
    - Consequences: Inability of users to rent available jeeps, loss of customer accounts and money
    - Work-around: Implement backup storage systems
- Dysfunctional Authentication System
    - Likelihood: Rare
    - Severity: Major
    - Consequences: Users can access private data
    - Work-around: Verify and encrypt login details and passwords
- Insufficient Renting System
    - Likelihood: Unlikely
    - Severity: Major
    - Consequences: Users cannot rent jeeps and application has no value
    - Work-around: Provide rental services by pen and paper

Dysfunctional Money System

- Likelihood: Unlikely
- Severity: Major
- Consequences: Company and users may lose money, no insurance profit
- Work-around: Take physical cash