Requirements Definition

*Big Blue’s Parking Genie*

1. **Introduction and Context**

*Finding convenient parking locations can be difficult when attending large university sponsored events, including sporting events. The ease at which one obtains and uses their parking spot may ruin or enrich such experiences. For example, consider attendance of an Aggie football game. Some parking is available at the Maverick Stadium parking lot but depending on arrival time it may quickly become scarce. Other parking lots are available around the university, but their locations may be less clear, especially to those unfamiliar with campus. The location of and navigation to suitable parking spots can be a stressful experience. This stress is only compounded by other factors such as heavy traffic, cost of the spot, and distance from the event. Big Blue’s Parking Genie (Genie) intends to make parking a more positive experience at Utah State University by collecting the universal parking experience into a single tool.*

*This document details the user goals, requirements, and features of the Genie web application. The aim of Genie is to ease the discovery, purchase, and access of parking spots for patrons of Utah State University events. These patrons may be public customers, students, or others who attend events. Additional users of Big Blue’s Parking Genie include university personnel, parking lot owners, and parking attendants. Section 2 of this document outlines the goals and process of these users. Section 3 details the requirements necessary for the app to function as intended. Section 4 introduces additional non-functional requirements to the application. The Genie development team will deliver the initial deployment before May 2021. Once complete, Genie should be able to assist a variety of people using either desktop or mobile devices. Potential future features and a glossary are included in sections 5 and 6 respectively.*

1. **Users and their Goals**

*This section contains identifies of the users of the proposed system and their goals, illustrated and supported by Use Case diagrams. Here “users” is a board term that could include other software systems.*

1. **Functional Requirements**

*This section contains requirements or constraints on the functioning of the proposed system, written in way that doesn’t pre-suppose “how” the system will accomplish those requirements. The requirements should be organized in hierarchy of increasing specificity and presented in outline form so they are easy to reference. Each requirement must stand on its own because it could be referenced or quoted in other documents without the benefit contextual information. Where appropriate, the requirements could include statements about of the rationale (motivation) and/or priority (importance to the client.)*

1. **Non-functional Requirements**

*This section contains requirements that describe and/or constrain the development process. For example, a non-functional requirement may state that development will follow an Agile method with weekly iteration meetings.*

1. **Future Features**

*This section contains a list of ideas or features that are beyond the scope of the project.*

1. **Glossary**

*This section contains a list important terms and their definition.*

TESTING NEW GIT REMOTE COLLABORATOR ACCESS