

Golden Eagle-Parking System (GE-PS)

CS 3337 Software Engineering

Functional Requirements and Design Document (FRD/SDD)

Prepared by:

Alvarez, Angie
Castanedo, Phoebe
Diaz-Lopez, Kevin

May 2020

Golden Eagle-Parking System (GE-PS)

Prepared By:

Kevin Diaz-Lopez
Angie Alvarez
Phoebe Castanedo

Approved By:

Jose Macias

May 2020
Date

Richard Cross

May 2020
Date

Document Change Log

Update	Date Released	Changes
Draft #1	3/14/2020	Delivery of the Software Requirements document
Draft #2	4/15/2020	Update to all sections
Final Document	x/xx/2020	Last update and upload to CSNS

List of TBD Items

Page	Item	Description	Status
Section 2	Completion and corrections	To be completed during the 2nd delivery of the document	Completed in April 2020
Section 4	Modules Code	To be completed before final delivery	To be delivered in May 2020

Table of Contents	Page
1.0 INTRODUCTION	7
1.1 Purpose	7
1.2 Scope	7
1.2.1 Document Organization	8
1.2.2 Relationship to Other Documents	8
1.3 Architecture	8
1.3.1 Context Diagram (Data Flow Diagram, DFD Level 0)	8
1.3.2 Project Description and Major Functions	9
1.3.3 Hardware and Software Considerations	9
1.4 Documentation of the Development Process	9
1.5 References	10
1.5.1 Controlling Documents	10
1.5.2 Applicable Documents	10
1.5.3 Standards	10

2.0	DETAILED FUNCTIONAL DESCRIPTION OF GE-PS.....	11
2.1	Detailed Functional Description	11
2.1.1	Data Flow Diagram (DFD) Level 1	11
2.1.2	Detailed Functional Description of Major Sub-Units	12
3.0	GE-PS FUNCTIONAL REQUIREMENTS	13
3.1	Detailed Functional Requirements	13
3.2	Non-Functional Requirements	15
3.3	Hardware Requirements	15
4.0	SOFTWARE IMPLEMENTATION.....	16
4.1	Introduction.....	16
4.2	Implementation.....	16
A.	ACRONYMS	18
B.	DATA DICTIONARY	19

1.0 INTRODUCTION

1.1 Purpose

The purpose of this document is four-fold:

1. Define a full set of requirements for each of this project's compositional modules. These sections correspond to a Software Requirements Document, SRD. The requirements are to be used in the implementation (coding) of the software written as part of this project.
2. Define the design for this project. These sections correspond to a Software Design Document (SDD). The design is to be used as a basis for the DFD level 0 and level 1 (DFD 0 and DFD 1).
3. Document the implementation (coding) of this project selected major modules. These sections correspond to a Software Implementation Document (SID). (Time constraints may allow only a partial level of completion).
4. Serve as the basis for the charts to be used during this project oral presentation.

1.2 Scope

This documentation is developed as part of the cs3337 project. The Implementation (Section 4) corresponds to a selection of these requirements.

The scope of this document includes the following:

- **GE-PS Project** functional and non-functional requirements. These requirements are organized by the key functional units shown on the Level 1 DFD given in section 2.0.
- A trace matrix, relating functional requirements to functional units as described in the DFD 1. Higher level DFDs will not be included/required for this project.
- General descriptions of the hardware that may be necessary for implementation of the project.

1.2.1 Document Organization

The organization of this document is in sections geared to provide a natural 'flow' or allocation of requirements to each unit included in the DFD level 1 shown in section 2.

1.2.2 Relationship to Other Documents

This is a self-contained document. Relationship to other documents in the literature, if any, is given in sub-section 1.5.

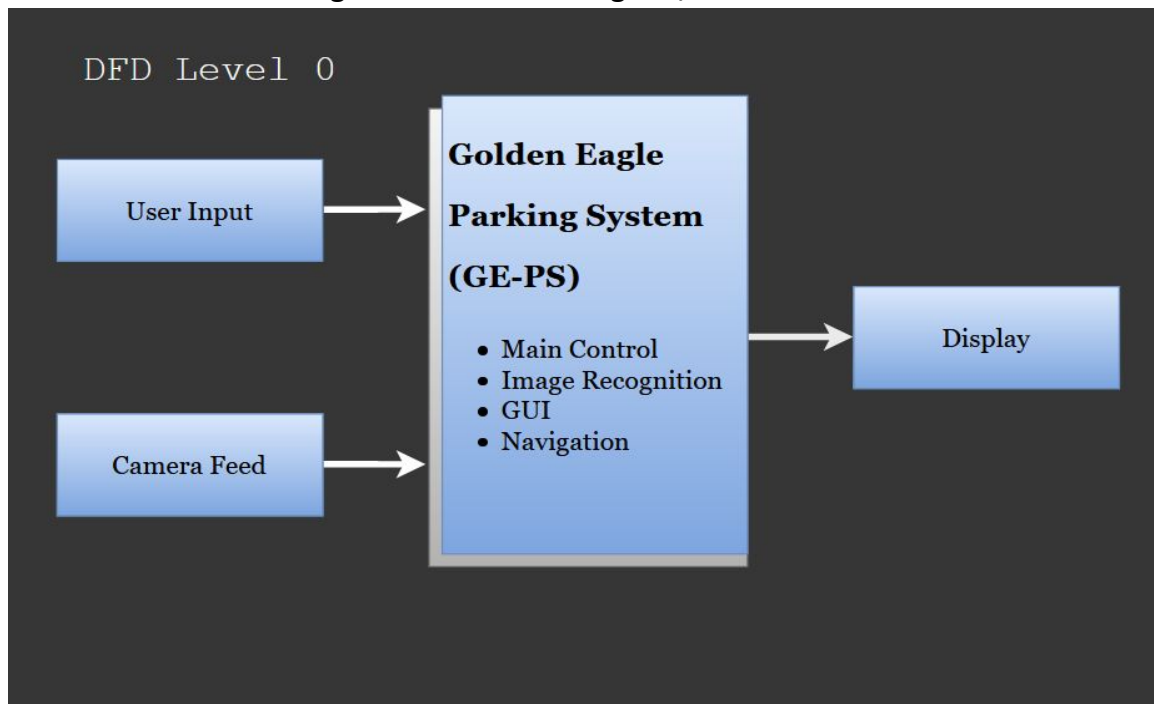
1.3 Architecture

1.3.1 Context Diagram (Data Flow Diagram, DFD Level 0)

This project overall design/architecture is summarized in the Context Diagram (DFD Level 0) shown below. A more detailed Functional Description, DFD Level 1, is given in the following section, Section 2.

This figure identifies at the highest level all inputs and outputs plus this project's major units described in its DFD Level 1.

Figure 1-1: Context Diagram, DFD Level 0



1.3.2 Project Description and Major Functions GE-PS

The purpose of this project is to streamline the parking process at Cal State LA and provide additional accessibility to visitors, students, and faculty of the campus. The app provides users with a clear guide to the available parking spaces on campus, and also lets users know which parking space is the most convenient for them based on the information provided about where on campus they need to go.

To this end, GE-PS is displayed primarily as a map. Each of its functions are intended to streamline visits to CSULA in that visitors do not waste time searching for parking spots.

1.3.3 Hardware and Software Considerations

This project requires standard computer/cellphone hardware as well as cameras. The cameras will be set up to provide a full view of the parking lot and those images will be processed through image recognition. Cellphones will be used to display the GUI to the user and take user input.

1.4 Documentation of the Development Process

This project's detailed functional description is documented in section 2.0. Section 2 is a succinct software description document. The overall detailed functional description is based on DFD 1 which will be included in this document (DFDs above level 1 are not included due to time constraints). All major functional units included in DFD 1 are described in detail in this part of the document.

Requirements for the project are captured in Section 3.0. This section includes both functional and non-functional software requirements. Also, if needed, these requirements are supplemented with hardware requirements.

Section 4, Implementation/Coding, is the basis for the oral presentation which is part of the project work of this class.

1.5 References

The references used in the creation of this document are listed below.

- Cs3337: Document Template

1.5.1 Controlling Documents

There is no controlling document for this FRD/SDD.

1.5.2 Applicable Documents

No additional applicable document has been used in the production of this document.

1.5.3 Standards

No standard has been used in the creation of this document. However, some Standards described in textbooks have been examined as a reference. In particular, the IEEE standard was briefly discussed and examined.

2.0 DETAILED SOFTWARE DESCRIPTION OF GE-PS

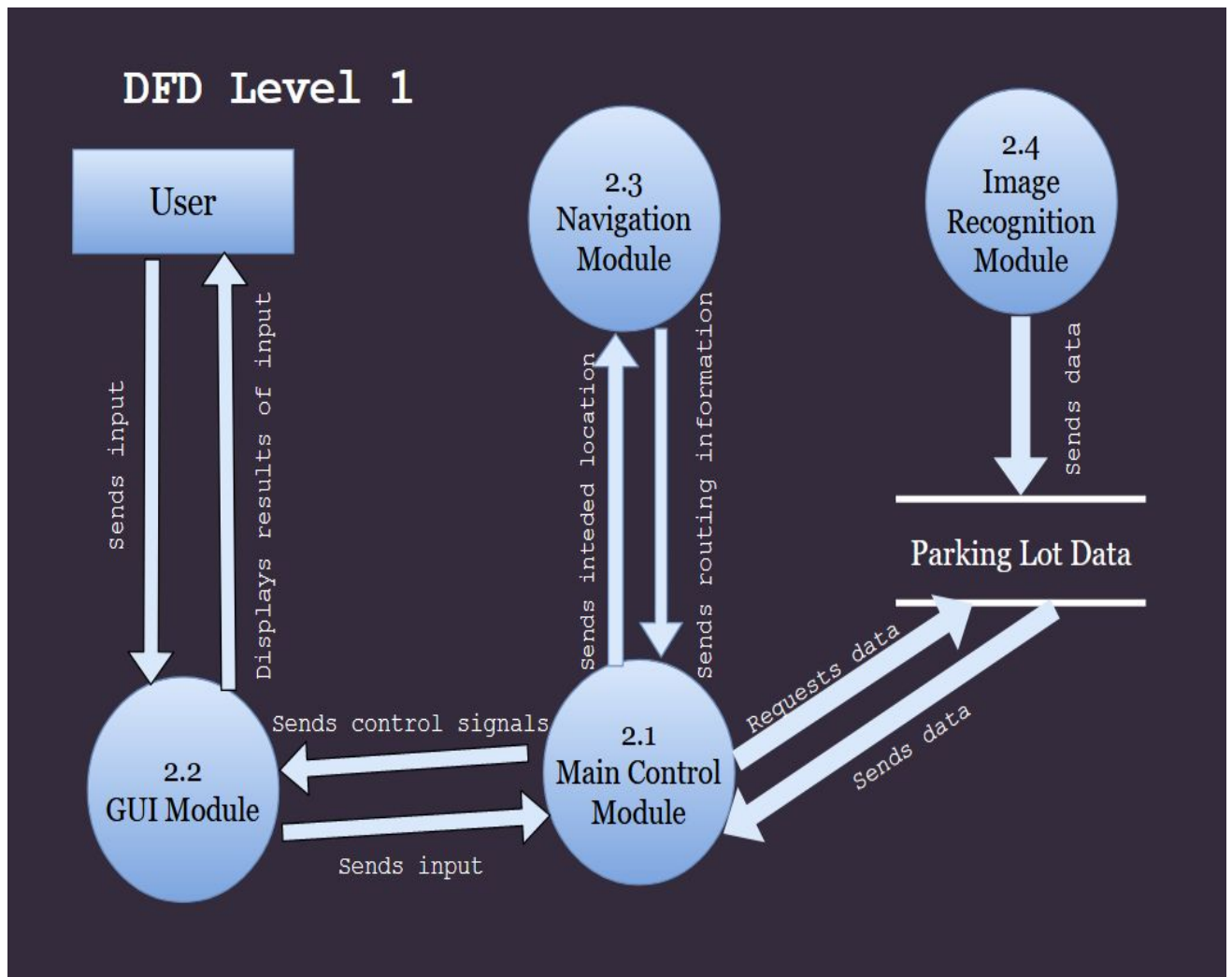
2.1 Detailed Software Description

The major design tool used is the Data Flow Diagram, DFD. The rationale for its selection was its simplicity. Additional tools could have been used if a stronger correlation from Design to Requirement to Implementation would have been necessary.

2.1.1 Data Flow Diagram (DFD) Level 1

This project's major functional units are shown in the DFD Level 1 shown below.

Figure 2.1: Level 1 DFD



2.1.2 Detailed Functional Description of Major Sub-Units

The description of GE-PS's major functional units shown in Figure 2.1 follows. Most of GE-PS's operations depend on the use of cameras to ensure accurate availability of parking spots.

Main Control - Module 2.1

The Main Control Module (MCM) is responsible for the exchange of information and data between all modules such as the availability of parking spots and user location.

GUI - Module 2.2

The GE-PS GUI receives all user input and provides a visual menu of all features of the GE-PS app.

The GUI on a user's phone allows access to features such as directions to campus (using another GPS app), directions around campus, parking space availability, input of class schedule or a building that the user is visiting, and viewing a model of the CSULA campus. The GUI will use the touchscreen of a phone to navigate through the app's menu.

Only once the user is within range of Cal State LA will directions around campus become accessible. When in range, the app will be able to guide the user from one end of the campus to another, in the event that an open space in a distant parking structure is recommended in response to the user's input to be provided with the most convenient space available based on their destination.

The parking space availability will be supplied by the database, which in turn is updated by the Image Recognition Module. Frequency of requests for this data is determined by the user.

Navigation - Module 2.3

The Navigation Module provides users with routing to the available parking space using their device's internal GPS. Additionally, if desired by the user, it can determine which parking space is closest to the user's intended location, provided that information, making visitation more efficient and convenient.

Image Recognition - Module 2.4

The Image Recognition module is responsible for post-processing and interpretation of the feed received from the cameras. Once processed into a boolean array, this module sends it to be stored in the parking spot database. This database may only be altered by this module.

3.0 GE-PS FUNCTIONAL REQUIREMENTS

3.1 Detailed Functional Requirements

This Section lists this project functional requirements. All requirements closely relate to the design modules described in Section 2 and pictured in the corresponding DFD 1. This section includes the complete set of functional requirements.

Since this project, (due to time limitation) did not require the "Requirement Analysis" major activity, some requirements have included additional explanations (rationale) for cases in which the statement of the requirement was deemed insufficient or required additional clarification.

There is a direct correlation between the design modules and the requirements which makes requirements organization more consistent. For example, requirement number "n" affecting module 2.1 will be labeled 3.1.n.

Module 2.1: Main Control Module (MCM)	
Requirement No.	Requirement Description
3.1.1	The Main Control Module shall receive user input through the GUI Module (module 2.2).
3.1.2	The Main Control Module shall send the user's intended location information to the Navigation Module (module 2.3).
3.1.3	The Main Control Module shall receive the recommended destination information from the Navigation Module (module 2.3).
3.1.4	The Main Control Module shall request and receive information from the parking lot data.
3.1.5	The Main Control Module shall send the collected data to the GUI Module (module 2.2) for display.

Design Module 2.2: Graphical User Interface (GUI)	
Requirement No.	Requirement Description
3.2-1	GUI shall receive all user input.
3.2-2	GUI shall allow users to optionally input their class information/destination on campus.
3.2-3	GUI shall send the user's input to the Main Control Module (module 2.1) to be routed to Navigation Module (module 2.4).
3.2-4	GUI shall interpret the arrays to provide a map display of each parking structure upon request.
3.2-5	GUI shall display a map of parking lots with distinct indications of vacant and occupied spaces.
3.2-6	GUI shall be able to 'tag' the user's space so there is a clear indication of where they are parked.
3.2-7	GUI shall request space array updates from Main Control Module (module 2.1).

Design Module 2.4: Image Recognition Module (IRM)	
Requirement No.	Requirement Description
3.4-1	The Image Recognition Module shall receive and interpret the video camera feed.
3.4-2	The Image Recognition Module shall be able to distinguish between an occupied parking space and a vacant parking space.
3.4-3	The Image Recognition Module shall translate the parking spaces from the video feed into a boolean array.
3.4-4	The Image Recognition Module shall send the generated arrays to be stored in a database that is sorted by parking structure.

Design Module 2.3: Navigation Module (NM)	
Requirement No.	Requirement Description
3.3-1	The Navigation Module shall receive the user's schedule from the Main Control Module (module 2.1).
3.3-2	The Navigation Module shall recommend the parking lot closest to the user's destination on campus.
3.3-3	The Navigation Module shall store the user's schedule persistently even when the app is closed.
3.3-4	

3.2 GE-PS Non-Functional Requirements

This Section collects this project Non-Functional Requirements. All non-functional requirements are numbered "NF – n" where "n" indicates the nth requirement.

- NF - 1 GE-PS requires internet access so the app may receive updates to parking availability from the externally stored arrays.
- NF - 2 The GE-PS application must require no technical input so it may be used by any CSULA visitor.

3.3 Hardware Requirements

SRD for the Golden Eagle-Parking System, GE-PS, by Alvarez, Castanedo, and Diaz-Lopez Spring 2020.

This Section collects all system hardware requirements. All hardware requirements are numbered “H – n” where “n” indicates the nth requirement.

- H - 1 The front-end, user-directed application of GE-PS will be run on smartphone hardware.
- H - 2 For parking space data to be accurately collected, GE-PS requires cameras to be situated such that their feeds have in view all parking spaces to be monitored. If adequate lighting is not always available then these cameras must have night vision capabilities.
- H - 3 For the Navigation Module (Module 2.4) to display routes and choose parking spaces, the device’s GPS must be enabled.

4.0 PROJECT IMPLEMENTATION

4.1 Introduction

In this section (some of) the modules designed in Section 2 with requirements listed in Section 3 will be implemented using a pre-selected programming language. Actual code will be provided in a “zip” file. Each module is implemented (coded) in correspondence with the design sections defined in section 2 and responding to the requirements listed in its correlated sub-section in chapter 3.

4.2 Implementation

4.2.1 Turn on/off all the lights (voice command)

```
"turn :onOrOff all the lights": function(onOrOff) {  
  
    if (userTalking.priority > 4) {  
        if (onOrOff === "on") {  
            // alert("Turning on all the lights.");  
            $.each($scope.house.rooms, function(room) {  
                $scope.house.rooms[room].light.on = true;  
            });  
            $scope.house.$save();  
        }  
    }  
}
```



```

        responsiveVoice.speak("Turning on all the lights.");

    } else if (onOrOff === "off") {
        // alert("Turning off all the lights.");
        $.each($scope.house.rooms, function(room) {
            $scope.house.rooms[room].light.on = false;
        });
        $scope.house.$save();
        responsiveVoice.speak("Turning off all the lights.");
    }
    } else { responsiveVoice.speak(userTalking.name + ". You have no authoritah!
Must be 5.");
    }
    },

```

4.2.2 Set room temperature (voice command /w audio conformation)

```

"(set) (change) (make) (turn) (the) temperature (to) :deg (degrees)":
function(deg) {

    if (userTalking.priority > 2) {
        if (parseInt(deg) > 54 && parseInt(deg) < 101) {
            $scope.house.rooms[currentRoom.getAttribute("DBid")].thermostat.on =
true;
            $scope.house.rooms[currentRoom.getAttribute("DBid")].thermostat.temp =
deg;
            $scope.house.$save();
            responsiveVoice.speak("Set the temperature to " + deg + " degrees.");

        } else {
            responsiveVoice.speak("Temperature range must be between 55 to 100
degrees.");
        }
    } else {
        responsiveVoice.speak("You have no authoritah! Must be 3 or higher.");
    }
    },

```

4.2.3 order food (voice command /w audio conformation)

```
"order :number :units of *item": function(number, units, item) {  
  responsiveVoice.speak("Ordered " + number + " " + units + " of " + item);  
  
  fridgeRef.push().set(  
    {name: item, quantity: number, units: units}  
  );  
},
```

A. ACRONYMS

MCM Main Control Module

IRM Image Recognition Module

NM Navigation Module

B. DATA DICTIONARY