SURE PARK SYSTEM

Software Design Document

Abstract

The document describes the software design of a parking facility management system designed for Geoff's Transportation and Parking Service called the Sure Park System. The Sure Park System aims to efficiently utilize parking space, reduce traffic congestion and accidents, efficiently utilize personnel and reduce driver anxiety and frustration inside parking facilities.

Team 6

eBusinessConsultants

TABLE OF CONTENTS

Introduction

Purpose

This document will outline in detail the software architecture and design for the Sure Park Parking Management System. This document will provide several views of the system's design in order to facilitate communication and understanding of the system. It intends to capture and convey the significant architectural and design decisions that have been made for the Sure Park System.

Scope

This document provides the architecture and design of Release 1.0 of the Sure Park System. It will show how the design will accomplish the functional and non-functional requirements detailed in the Sure Park System Software Requirements Specification (SRS) document.

Intended Audience

This document is written on a technical level to address the technical department i.e. developers, managers and other stakeholders responsible for manifesting the Sure Park System.

Definitions & Acronyms

SPS: Sure Park System

AIA: Advanced Industrial Automation

JSP: Java Server Pages

MVC: Model-View-Controller

QR: Quick Response

GTPS: Geoff's Transportation and Parking services

FOS: Facility Operation Supervisor

System Overview

System Context Diagram

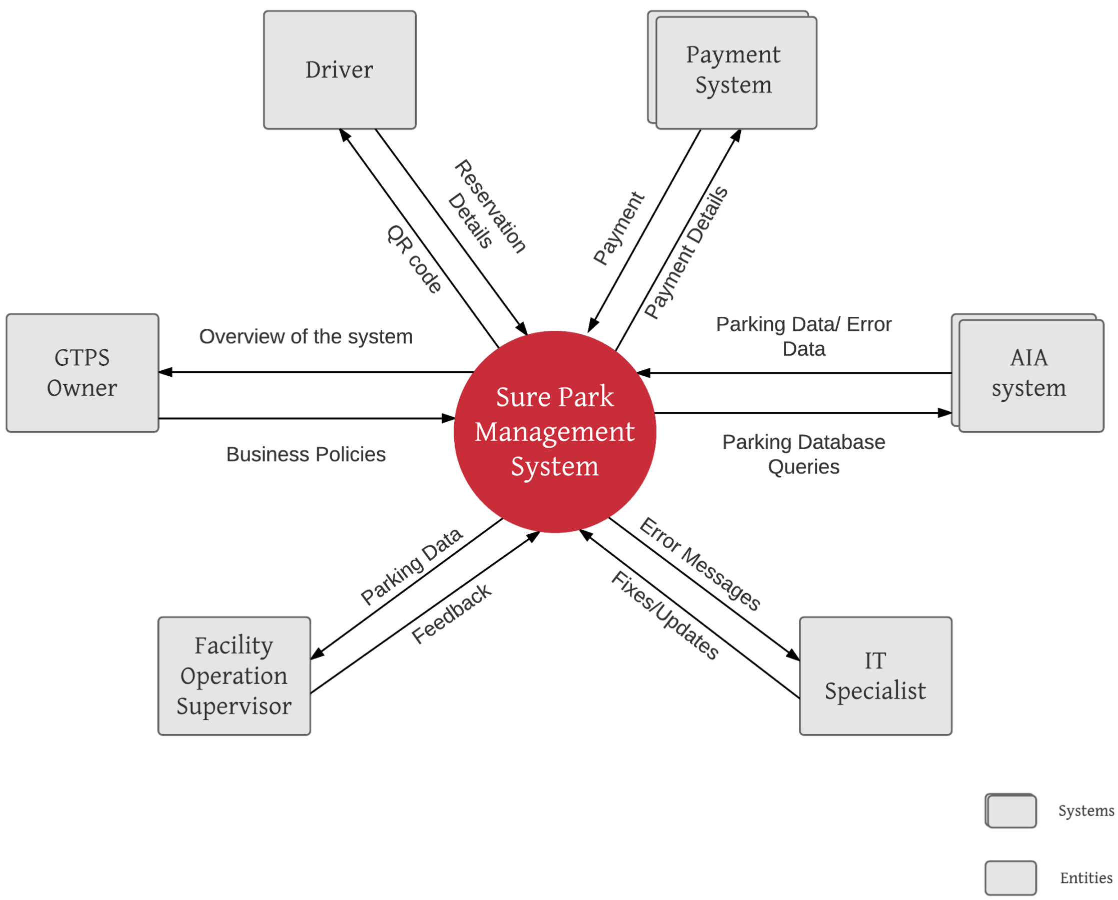


FIGURE 1 SYSTEM CONTEXT DIAGRAM

The above figure displays a bird's eye view of the data flow transpiring in the Sure-Park system. The different entities interact with the system by exchanging data pertinent to their functionalities.

The Sure-Park management system encapsulates the functionalities of its different modules which are:

- An application that enables drivers to register for and reserve parking spaces in any of the parking facilities owned by the customer and authorizes them for entry into the parking facilities.
- Facilities Management Console Application that enables the Facilities Operations Supervisors to oversee the operations of the parking facility they manage and charge penalty to the driver in case of wrong parking.
- Entry access and management applications that authorized drivers and allows them to enter the parking facilities using a QR code.

- Payment management systems that securely takes driver payment information, processes payment and generated receipt.

Stakeholder Introduction & Interaction

The interactions of different entities with the Sure-Park system are as follows:

- Driver: The driver is the user of the parking garage and interacts with the Sure-Park system to reserve a parking spot via a client app.
- Payment System: The payment system interacts with the Sure-Park system servers to help seamless processing of online payments.
- AIA System: The Sure-Park system servers queries the already implemented AIA system databases to obtain relevant information about the state of parking spaces and gates in the parking lot.
- IT Specialist: The IT specialist intercepts errors in the system, maintains the system and scarcely fixes bugs in the system.
- Facility Operation Supervisor: The FOS uses the Sure-Park system console to supervise and manage the parking lot, informing of error cases to the user/driver.
- GTPS Owner: The Sure-Park system enables the GTPS owner to impose business policies like penalty charges etc. on the users. The GTPS owner also receives an overview of the system to help in decision making and strategy planning.

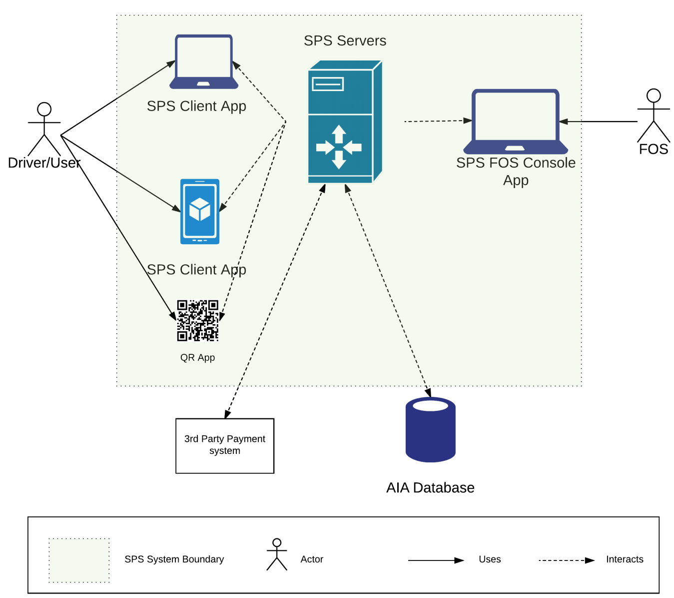


FIGURE 2 STAKEHOLDER INTERACTION WITH SPS

Architectural Drivers

Architectural drivers are formally defined as the set of requirements that have significant influence over architecture. In other words, these are some requirements that will help decide which structures to pick for system design and others that are less consequential in the context of software architecture. Architectural drivers shape the decision making process while designing the software. The design rationale is dependent on the architectural drivers.

High-level functional Requirements

Allow the driver to make a reservation

Description: After the user logged in, the user shall be able to book a parking spot for a particular date and time duration if the spot is not already booked.

Check if there are parking spaces available at certain facility

Description: After user logged in, let user choose one facility which has available parking lots. This function should be used when reservation and parking facility suggestions.

Generate an identification ticket number (QR code)

Description: After driver make a reservation, a QR code is generated and sent to driver.

Allow user to manage his/her account.

Description: After user logged in, he/she can change his/her name, email address, phone, driver's license, license plate # and payment methods.

Verify the login username and password.

Description: The application should make sure that username and password are correct.

Validate the payment when registration.

Description: Check the credit card information is valid or not.

Allow driver to establish monthly contracts.

Description: The registration system should allow drivers to establish monthly contracts with specific parking facilities. The QR code generated should be valid for one month.

The console shall verify and contact the owner of the vehicle if needed.

Description: if contact with drivers is necessary in some situation, the console should be able to verify the drivers and contact them.

The console shall see the availability of parking spaces in real time.

Description: the console should know the exact number of the availability in real time.

The console shall fetch the information about the improper parking from SPS system.

Description: the console should request the information about the improper parking in parking spots.

THE CONSOLE SHALL KNOW THE OVERSTAY VEHICLES THAT ARE NOT CONTRACT USERS AND INFORM AUTHORITIES.

Description: the console will have the information of the overstayed vehicles and obtain the driver's profile by license plate.

THE CONSOLE SHALL BE ABLE TO CONTROL THE GATE REMOTELY.

Description: the console will open/close/lock both the entry and exit gates in case of the malfunction of entry or exit system.

INTERFACE WITH THE ENTRY ACCESS SYSTEM

Description: The system shall interface with the entry access system to charge the driver for the duration of the stay.

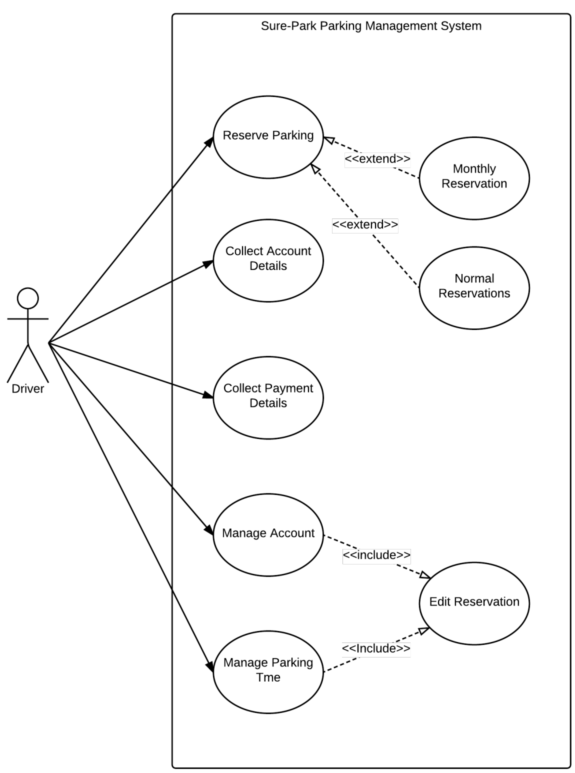
QR CODE VERIFICATION MACHINE.

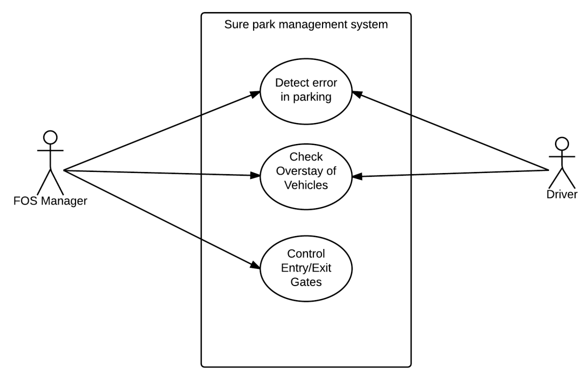
Description: The system shall have a QR code verification machines near the lift gates, which can be used to verify users' QR codes.

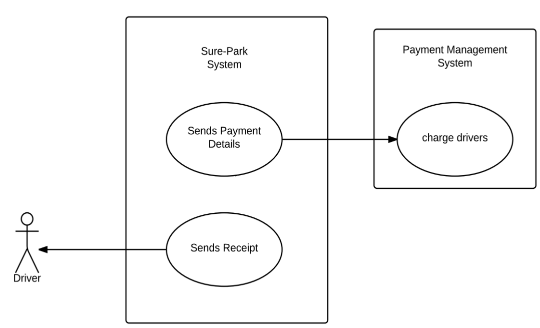
CONNECT PAYMENT SYSTEM TO THE SURE-PAYMENT SYSTEM.

Description: The system shall be connected to the user applications, payment system and supervisor console.

USE CASE DIAGRAMS







High-Level Design

Static Perspective

The aim of the static perspective is to observe the system when it is not compiled.

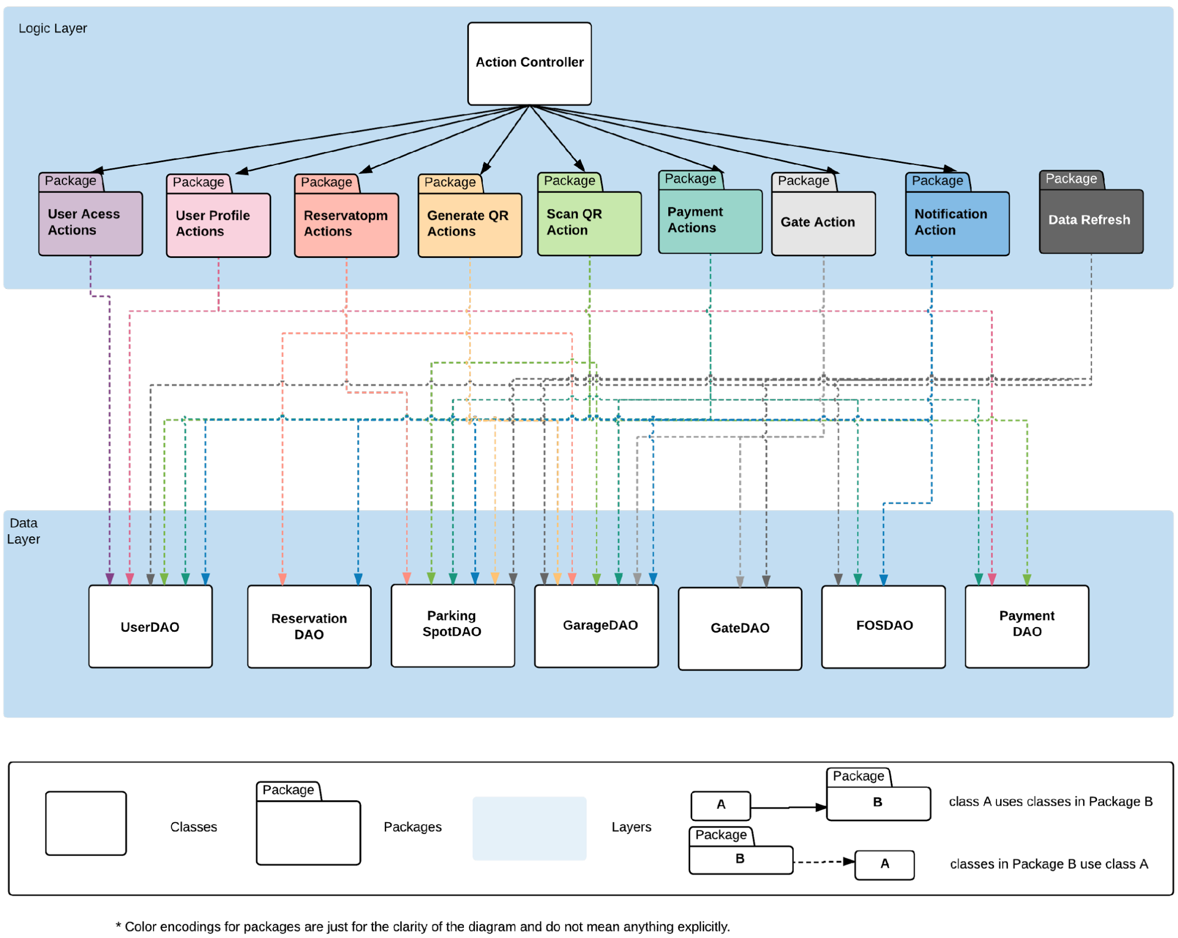


FIGURE 3 STATIC STRUCTURAL DIAGRAM

Design Rationale

(For description of packages and java classes, please refer to java class diagram design rationale)

Table 1 Element Responsibility Catalogue for the Architectural Design Decomposition

Associated Drawings: Figure 3	**Perspective: Static**
Element	Responsibilities

Logic Layer	Handles and process requests from client
Data Layer	Perform create, read, update, delete functions to the underlying database

Table 2 Relationship Responsibility Catalog for the Architectural Design Decomposition

Associated Drawings: Figure 3	**Perspective: Static**
Relationship	**Responsibility**
A $\dashrightarrow$ B	This relationship indicates that java classes in package A uses java class B
A $\longrightarrow$ B	This relationship indicates that java class A uses a java class in package B

Dynamic Perspective

The dynamic perspective aims at describing the system when it is compiled or it is running.

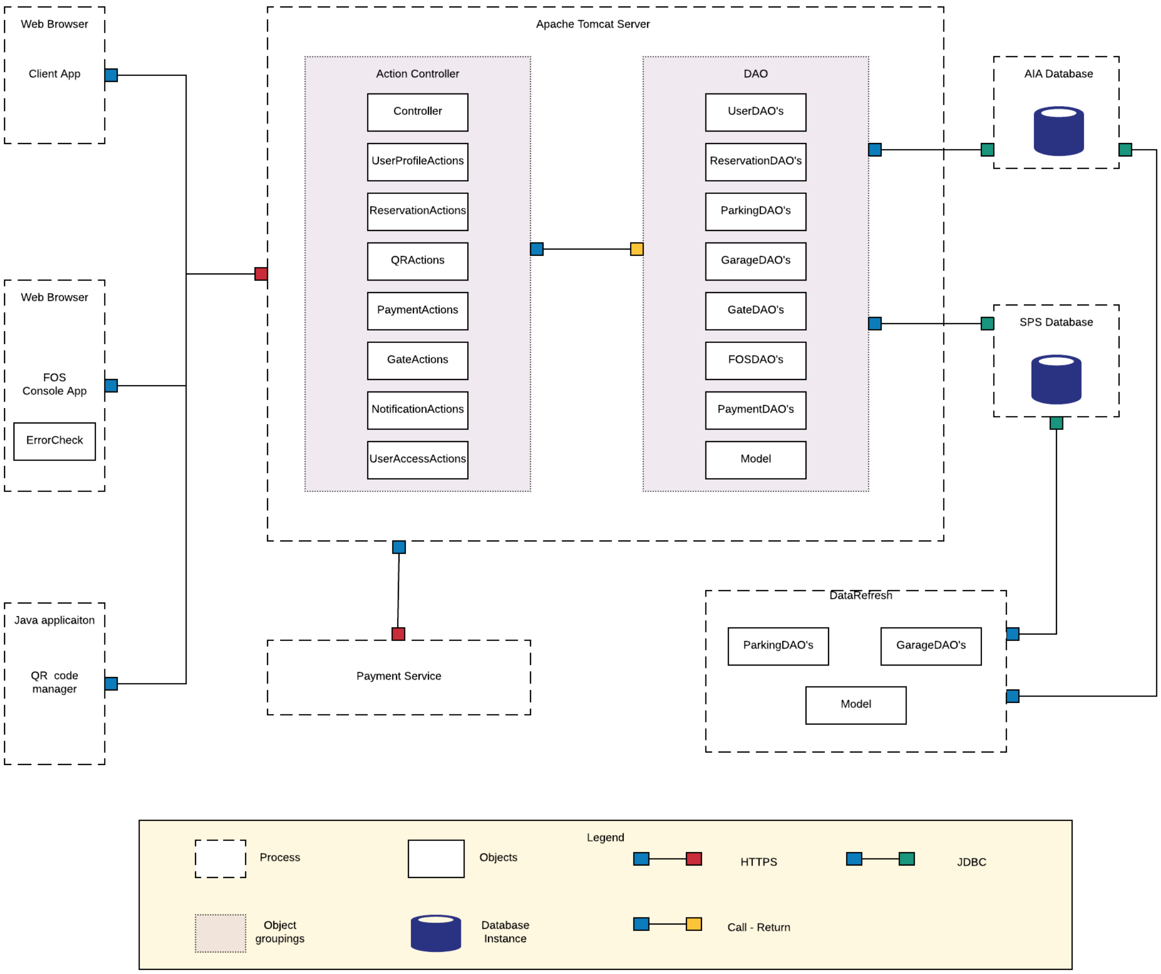


FIGURE 4 STRUCTURAL DYNAMIC DIAGRAM

Design Rationale

Table 3: Element Responsibility Catalogue for the Architectural Dynamic Design Decomposition

Associated Drawings: Figure 4	**Perspective: Dynamic**
Element	**Responsibilities**
Client App	Allows drivers to use the Sure-Park system in the web browser.

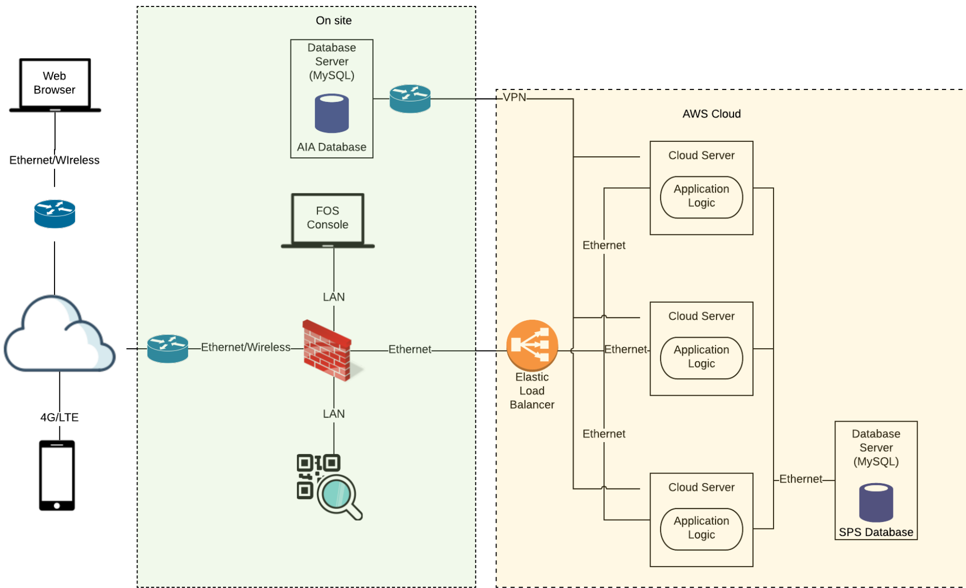
FOS Console App	Allows Facility Supervisors to manage the parking lot in the web browser.
QR Code Manager	Charge drivers parking fee at exit gate by scanning QR code.
Action Controller	Handles HTTP requests from client app, FOS console app, and QR Code Manager, and sends back HTTP response.
DAO	Provides SPS and AIA Database access.
Data Refresh App	Query AIA database for the available slots and parking error every 5 seconds.
Payment Service	Charge driver's credit/debit card for the parking fee.
AIA Database	Stores parking lot and gate data. Responds to database queries.
SPS Database	Stores user, reservation, payment, parking lot, FOS data etc. Responds to database queries.

Table 4: Relationship Responsibility Catalogue for the Architectural Dynamic Design Decomposition

Associated Drawings: Figure 4	**Perspective: Dynamic**
Relationship	**Responsibility**
HTTPS	Allows client and server to communicate with security
JDBC	Allows Tomcat Server to query the database server
Call-Return	Allows Action Controller, DAOs, and Data Refresh app to call each other and retrieve data.

Physical Perspective

The physical perspective describes the physical components of the system.



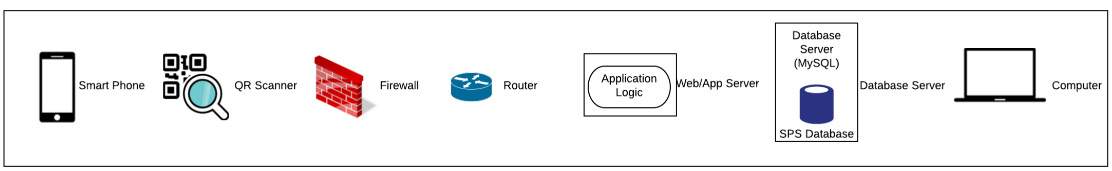


FIGURE 5 PHYSICAL DIAGRAM

Design Rationale

Table 5: Element Responsibility Catalogue for the Architectural Physical Design Decomposition

Associated Drawings: Figure 5	**Perspective: Physical**
Element	**Responsibilities**
Customer Computer	Allows drivers to use the Sure-Park system.
Customer Smart Phones	Allows drivers to use the Sure-Park system.
Internet	Allows drivers to remotely connects to the Sure-Park Servers.
FOS Work Computer	Allows Facility Supervisors to manage the parking lot.
Firewall	Protects the company's LAN from malicious attacks

QR Scanner	Charge drivers parking fee at exit gate by scanning QR code.
Load Balancer	Redirects client requests to multiple web/app cloud servers to balance the load.
Web/App Cloud Server	Handles HTTP requests from clients, formulates and sends back HTTP response.
Database Server (AIA)	Stores parking lot and gate data. Responds to database queries.
Database Cloud Server (SPS)	Stores user, reservation, payment, parking lot, FOS data etc. Responds to database queries.
Router	Allows hardware to connect to the internet.

Table 6: Relationship Responsibility Catalogue for the Physical Architectural Design Decomposition

Associated Drawings: Figure 5	**Perspective: Physical**
Relationship	**Responsibility**
4G/LTE	Allows Smart Phones to connects to the internet.
LAN	Allows Facility Supervisor's work computers and QR scanners to connect securely to the servers
VPN	Protects sensitive data by adding security and privacy.
Ethernet/Wireless	Allows hardware to connect to the internet or the local area network.

APPENDIX A

ENTITY-RELATIONSHIP DIAGRAM

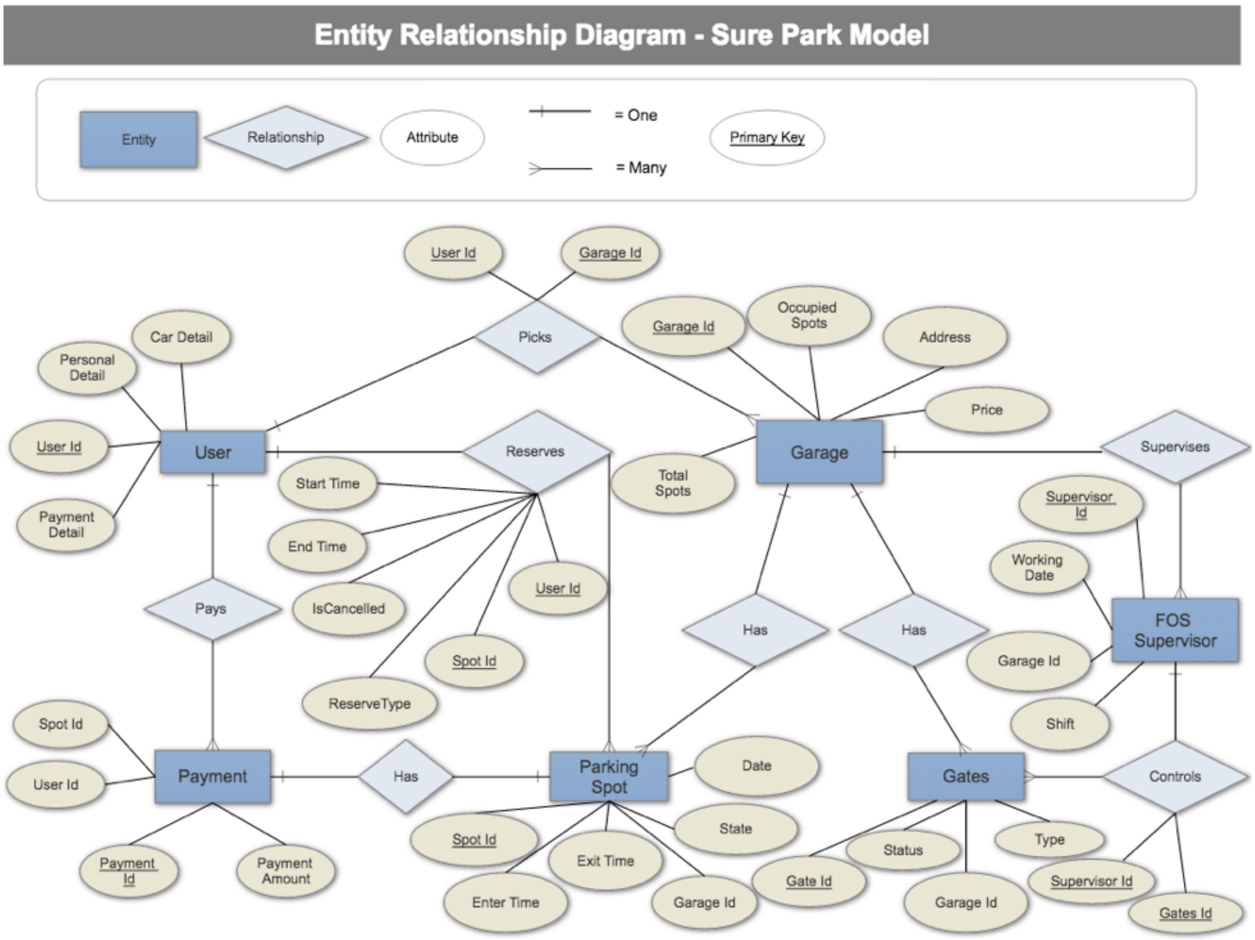


FIGURE 3 ER DIAGRAM

MYSQL SCHEMA

User					
Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	auto_increment
payment info	varchar(255)	YES		NULL	
personal info	varchar(255)	YES		NULL	
car info	varchar(255)	YES		NULL	
Picks					
Field	Type	Null	Key	Default	Extra
userId	int(11)	NO	PRI	NULL	
garageId	int(11)	NO	PRI	NULL	

Garage					
Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	auto_increment
occupiedSpots	int(11)	NO		NULL	
address	varchar(255)	NO		NULL	
Price	varchar(255)	NO		NULL	
totalSpots	int(11)	NO		NULL	
Payment					
Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	auto_increment
userId	int(11)	NO		NULL	
spotId	int(11)	NO		NULL	
paymentAmount	varchar(255)	NO		NULL	
Parking Spot					
Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	auto_increment
entryTime	varchar(255)	NO		NULL	
exitTime	varchar(255)	NO		NULL	
garageId	int(11)	NO		NULL	
state	varchar(255)	NO		NULL	
date	varchar(255)	NO		NULL	
Gates					
Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	auto_increment
status	varchar(255)	NO		NULL	
type	varchar(255)	NO		NULL	
garageId	int(11)	NO		NULL	
FOS staff					
Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	auto_increment
workingDate	varchar(255)	NO		NULL	
shift	varchar(255)	NO		NULL	
garageId	int(11)	NO		NULL	
Controls					
Field	Type	Null	Key	Default	Extra
gateId	int(11)	NO	PRI	NULL	
garageId	int(11)	NO	PRI	NULL	
Reserve					
Field	Type	Null	Key	Default	Extra

userId	int(11)	NO	PRI	NULL	
spotId	int(11)	NO	PRI	NULL	
startTime	varchar(255)	NO		NULL	
endTime	varchar(255)	NO		NULL	
isCancelled	varchar(255)	NO		NULL	
reserveType	varchar(255)	NO		NULL	

APPENDIX B

QUALITY ATTRIBUTE SCENARIOS

SECURITY

PRIORITY ASSERTED BY THE CUSTOMERS: 1 - High

REASON: If the driver information is disclosed, their payment details could be compromised and may lead to a lawsuit.

TECHNICAL RISK: 1 - Low.

REASON: To ensure the security of the system, an additional secure layer or encryption is needed on top of TCP/IP, which increases the complexity of the system and make it harder to maintain.

Raw quality attribute	System security
Stimulus	The driver's information is obtained by hackers who eavesdrops on the traffic
Source of the stimulus	Malicious hackers
Architectural elements	Payment transaction software
System response	The system makes the data unreadable for everyone except the sender and the receiver.
Response measures	The hacker cannot understand the encoded traffic between user and the system

PERFORMANCE

PRIORITY ASSERTED BY THE CUSTOMERS: 2 - High

REASON: If the system has bad performance i.e. long response times the confirmation of payment, confirmation of parking, QR code reading etc. flows are affected leading to bad user experience and congestion during entry/exit.

TECHNICAL RISK: Moderate

REASON: To improve the performance would require deploying redundant servers to handle the increased amount of requests quickly which would increase the budget of the project. Also, we would require time bound software mechanisms to make sure that response to requests are delivered timely, leading to increased complexity of the system.

Raw quality attribute	The system performance
Stimulus	Large number of requests the server. For example, reserving a parking spot, scanning a QR code, processing payment etc.

Source of the stimulus	All requests
Architectural elements	Client App, QR code app, FOS Console, Payment system
System response	The sure-park system uses replica servers and ensures time bound responses to requests.
Response measures	The sure-park system is able to generate response in less than 10 seconds to requests.

RELIABILITY

PRIORITY ASSERTED BY THE CUSTOMER: 1 - High

REASON: If the system fails while deployed then all ongoing transactions are lost leading to monetary losses and decreased credibility.

TECHNICAL RISK: 1 - High

REASON: The meant time to failure is depends on the minimum mean time of failure of the systems components. It is difficult to gauge mean time of failure for software component as they depend on end cases like race conditions which reveal themselves via rigorous testing. Hardware components generally have an assigned mean time to failure by the vendor.

Raw quality attribute	The system reliability
Stimulus	Race conditions, end cases for software components. Not monitoring the health of hardware components
Source of the stimulus	Hardware and Software components of the system.
Architectural elements	Sure Park System
System response	The system should anticipate failures via rigorous testing and health of hardware components should be monitored. In case of a failure the spare hardware components should be available and for software components personnel should be valuable to fix the bugs.
Response measures	Mean time to failure.

CONFIGURABILITY

PRIORITY ASSERTED BY THE CUSTOMER: 1 - High

REASON: The system if not configurable to accommodate itself to work in parking lots with different number of gates or different orientation of parking spaces then it would be difficult to deploy the system.

TECHNICAL RISK: Low

REASON: The system should be wary that the configuration be locked for each parking lot and cannot be changed by everyone. Also, the different configurations would require rigorous individual testing for each parking lot it is deployed.

Raw quality attribute	The system configurability
Stimulus	Different parking lots have different hardware setup and limitations.
Source of the stimulus	Different parking lots of the Geoff's Transportation and Parking Services
Architectural elements	The sure park system.
System response	The system has to accommodate to different parking lots.
Response measures	The SPS should be able to be deployed for a new parking facility in no more than 3 days.

SCALABILITY

PRIORITY ASSERTED BY THE CUSTOMER: 1 - High

REASON: The system should be able to handle and anticipate increased load on the system. It should also be easily able to accommodate the load of new parking facilities when added. A failure to do so will affect performance, availability and reliability of the system with increased load.

TECHNICAL RISK: Moderate

REASON: The essential technique for ensuring availability and reliability is redundancy and the overprovisioning of resources but these resources should be disparate from the resources needed to scale or it would affect availability, reliability and performance of the system.

Raw quality attribute	The system scalability
Stimulus	Increased number of request on the SPS.
Source of the stimulus	Different parking lots and of the Geoff's Transportation and Parking Services and users of the client application.
Architectural elements	Servers of the SPS.
System response	New resources can be added within seconds since the servers are deployed on cloud.
Response measures	The servers should anticipate the need for scalability and new resources dedicated to handle increased load should be added in a few minutes.

AVAILABILITY

PRIORITY ASSERTED BY THE CUSTOMER: 1 - High

REASON: High availability is a quality that aims at the indentured availability of a system during a certain period. If the system is unavailable for long periods of time it might lead to losses for the client. Availability of the system and Reliability of the system go hand in hand.

TECHNICAL RISK: Moderate

REASON: The essential technique for ensuring availability and reliability is redundancy and the overprovisioning of resources.

Raw quality attribute	The system availability
Stimulus	Failures in the system. For example, Server not working, Client app not working etc.
Source of the stimulus	Requests from components of SPS.
Architectural elements	Servers
System response	Redundant servers being deployed when the active server fails.
Response measures	The redundant servers should be deployed in less than 5 seconds in case of failure.

Assumptions

The Sure-Park system has the following underlying assumptions while covering as many scenarios as it can:

A1. The AIA system database always has accurate information about the parking spaces.

A2. There is a FOS supervising the parking lot at all times it is open to public.

A3. The AIA system always responds accurately to queries of opening and closing of the entry and exit gates.

A4. The user/driver enters accurate payment information into the system and the information maps to a valid payment source mapped to the user/driver.

A5. The 3rd party payment system employed always responds with appropriate turnaround times to the requests from the Sure-Park system.

A6. The AIA system securely handles database queries from Sure-Park system.

A7. The AIA system activates the alarm indicators at parking spots when driver incorrectly parks.

A8. Lastly, the customer has access to his or her email and a cell phone. However, it is not assumed that the customer will check his or her email frequently or that the customer has a smart phone capable of accessing web pages on the Internet or downloading apps.