Capgemini

High Level Design & Low Level Design

1. INTRODUCTION

1.1. Intended Audience

It is intended to be read by Customers.

1.2. Project Purpose

● The best thing about this application is it helps customers to request for household services.

● This application is compatible with almost every smart device having an internet connection.

● This application doesn’t require any installation process and provides the best options for home services.

1.3. Key Project Objectives

- Customer Id and Technician Id is displayed whenever connection is established.
- Best vendors are easily available on one click.
- Both Technicians and customers can change their basic information like address and phone number etc.
- The customers are also required to choose a password which they will use to avail the services in future.
- Admin of the system have control on customer, technician and service databases. He/she need to be authenticated through a password.

1.4. Project Scope and Limitation

This project aims for customers to avail the home services easily from any device having internet connection. The customer can easily access this application just by using their Customer Id and Password from any device.

This application doesn’t require any installation process and provides technicians who are skilled in various household services, it is compatible with almost every smart device.

1.5. Functional Overview

Following header files are included in the program:

- #include <stdio.h>
- #include <stdlib.h>
- #include <string.h>
- #include <ctype.h>

2. DESIGN OVERVIEW

2.1. Design Objectives

An Easily accessible and affordable options for millions of households without any problem.

2.2. Architectural Strategies

The strategy is to use a simple application which is easy to use and will work across multiple platforms.

2.2.1. Design Alternative

We have used three structure to store data i.e., Customer, Technician and Admin and we are using appropriate data types for all the structures.

2.2.2. User Interface Paradigms

The user interface is easy to use and handles errors effectively. And it also uses easy to understand sentences so that customer feels at home while using application.

2.2.3. Error Detection / Exceptional Handling

- If the customer gives invalid Customer Id and password, it shows an error message which indicates that either of the input is invalid.

2.2.4. Performance

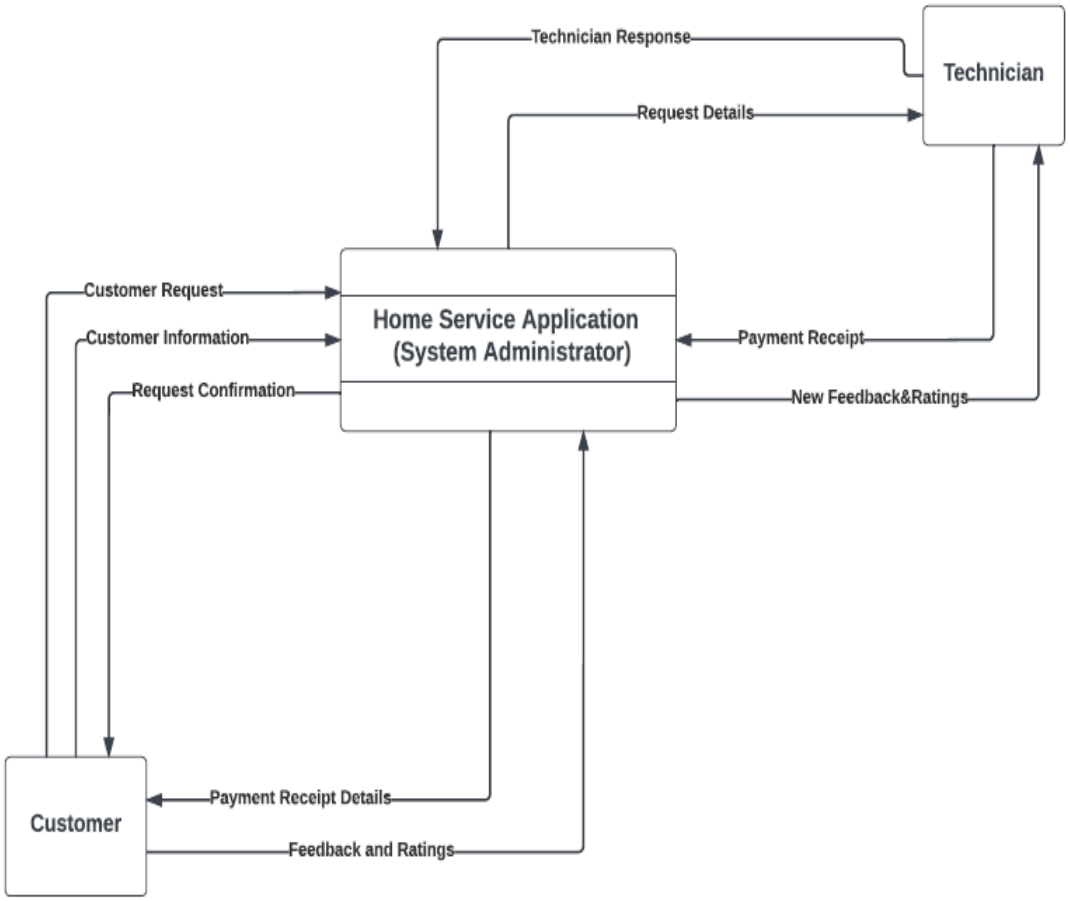
The system will work on the customers terminal as well as on cell phones.

2.2.5. Maintenance

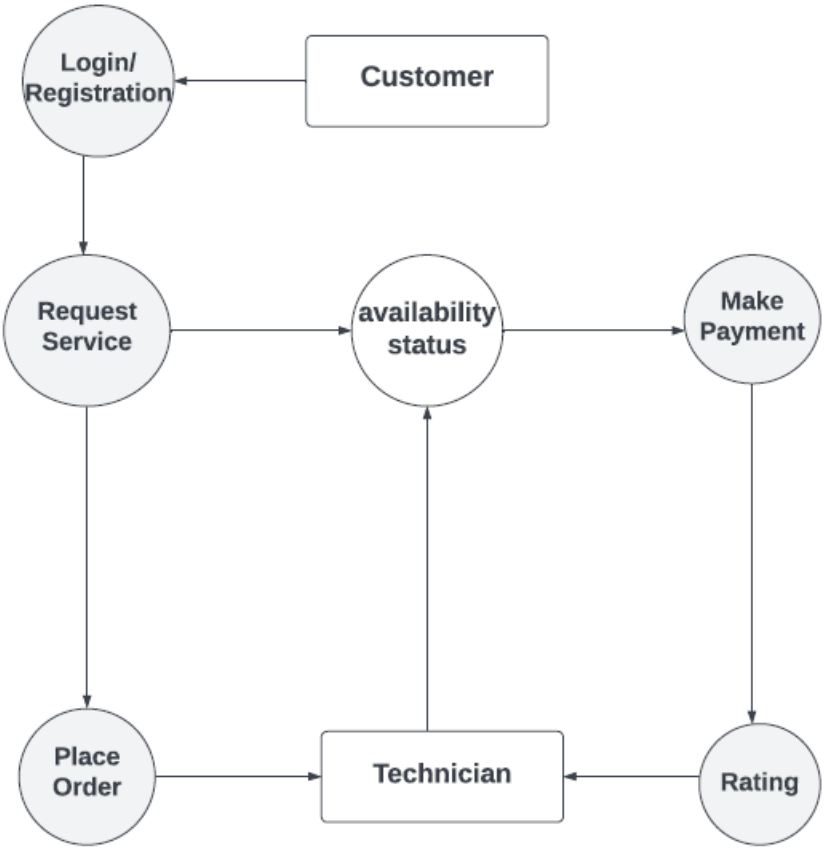
Very little maintenance would be required for this setup as it purely runs on C-language and Data Structures.

3. DETAILED SYSTEM DESIGN

3.1. DFD Level 0



3.2. DFD Level 1



4. ENVIRONMENT DESCRIPTION

4.1. Time Zone Support

IST-Kolkata

4.2. Language Support

English

4.3. User Desktop Requirements

● 64-bit processor, 1 GHz or faster

● At least 4 GB free hard drive space

● At least 1 GB RAM Server

4.4. Deployment Considerations

● Local storage is used

● No network latency to consider

● To scale buy a bigger CPU, more memory, larger hard drive, or additional hardware

4.5. Integration Requirements

● Language: - C

● Tools: - Valgrind, Makefile, Ctags, Cscope, Splint

● Complier: - gcc

● Linux Environment