CFSS\_v0.3

CALL FORWARDING SYSTEM SIMULATOR

The purpose of this document is to provide a template for documenting CFSS.

**Document Control :**

| **Project Revision History** | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  | |  |  |  |  |  |
| **Date** | **Version** | **Author** | **Brief Description of Changes** | | | | **Approver Signature** | |
| 12/11/2022 | 0.1v | Group04 | Initial Draft | | | |  | |
| 24/11/2022 | 0.2v | Group04 | Added Flowchart | | | |  | |
| 25/11/2022 | 0.3v | Group04 | Added Variables | | | |  | |

[**1.INTRODUCTION**](#_heading=h.3znysh7) **5**

[1.1.Intended Audience](#_heading=h.2et92p0) 5

[1.2.Acronyms/Abbreviations](#_heading=h.tyjcwt) 5

[1.3.Purpose of the project](#_heading=h.3dy6vkm) 5

[1.4.Key project Objectives](#_heading=h.1t3h5sf) 5

[1.5.Project scope and limitation](#_heading=h.2s8eyo1) 5

[1.5.1.In scope](#_heading=h.17dp8vu) 5

[1.5.2.Out of Scope](#_heading=h.3rdcrjn) 6

[1.6.Functional Overview](#_heading=h.26in1rg) 6

[1.7.Assumptions, Dependencies & Constraints](#_heading=h.lnxbz9) 6

[1.8.Risks](#_heading=h.35nkun2) 6

[**2.DESIGN OVERVIEW**](#_heading=h.44sinio) **6**

[2.1.Design Objectives](#_heading=h.2jxsxqh) 6

[2.1.1.Recommended Architecture](#_heading=h.hj2asmuaaljt) 7

[2.2.Architectural Strategies](#_heading=h.3j2qqm3) 7

[2.2.1.Design Alternative](#_heading=h.1y810tw) 7

[2.2.2.Reuse of Existing Common Services/Utilities](#_heading=h.2xcytpi) 8

[2.2.3.Creation of New Common Services/Utilities](#_heading=h.3whwml4) 8

[2.2.4.User Interface Paradigms](#_heading=h.2bn6wsx) 8

[2.2.5.System Interface Paradigms](#_heading=h.3as4poj) 8

[2.2.6.Error Detection / Exception Handling](#_heading=h.1pxezwc) 8

[2.2.7.Memory Management](#_heading=h.49x2ik5) 8

[2.2.8.Performance](#_heading=h.147n2zr) 8

[2.2.9.Security](#_heading=h.23ckvvd) 8

[2.2.10.Concurrency and Synchronization](#_heading=h.ihv636) 9

[2.2.11.Housekeeping and Maintenance](#_heading=h.32hioqz) 9

[**3.SYSTEM ARCHITECTURE**](#_heading=h.1hmsyys) **9**

[3.1.System Architecture Diagram](#_heading=h.41mghml) 9

[3.2.System Use-Cases](#_heading=h.2grqrue) 9

[3.3. Subsystem Architecture](#_heading=h.vx1227) 10

[3.4. System Interface](#_heading=h.3fwokq0) 11

[3.4.1.Internal Interfaces](#_heading=h.1v1yuxt) 11

[3.4.2. External Interfaces](#_heading=h.4f1mdlm) 11

[**4.DETAILED SYSTEM DESIGN**](#_heading=h.h3kkr49wihk) **11**

[4.1.Key Entities](#_heading=h.19c6y18) 11

[4.2.Detailed-Level Database Design](#_heading=h.hbalq3xqfua3) 13

[4.2.1.Flowchart](#_heading=h.28h4qwu) 13

[4.2.2 Level- 0 Diagram](#_heading=h.nmf14n) 14

[4.2.3 Level -1 Diagram](#_heading=h.1mrcu09) 14

[4.2.4 Level -2 Diagram](#_heading=h.46r0co2) 15

[4.3.Requirements Specifications](#_heading=h.2lwamvv) 15

[4.3.1.Register and Unregister](#_heading=h.111kx3o) 15

[4.3.2.Support three types of call forwarding](#_heading=h.3l18frh) 15

[4.3.3.Support normal call](#_heading=h.206ipza) 16

[4.3.4.Archival and retention requirements](#_heading=h.2zbgiuw) 16

[4.4. Disaster and Failure Recovery](#_heading=h.3ygebqi) 16

[4.5.Business Process Workflow](#_heading=h.2dlolyb) 17

[4.6.Variables](#_heading=h.sqyw64) 17

[4.7.Activity Diagram](#_heading=h.fj48ijlnnhp8) 18

[4.7.1.Sequence Diagram](#_heading=h.1rvwp1q) 18

[4.9. Data Migration](#_heading=h.2r0uhxc) 19

[**5.ENVIRONMENT DESCRIPTION**](#_heading=h.1664s55) **19**

[5.1.Time Zone](#_heading=h.3q5sasy) 19

[5.2.Language Support](#_heading=h.25b2l0r) 19

[5.3.User Desktop Requirements](#_heading=h.kgcv8k) 19

[5.4.Server-Side Requirements](#_heading=h.34g0dwd) 19

[5.4.1.Deployment Considerations](#_heading=h.1jlao46) 19

[5.4.2. Application Server Disk Space](#_heading=h.mh1hrpteu9zp) 19

[5.4.3.Database Server Disk Space](#_heading=h.2iq8gzs) 20

[5.4.4.Integration Requirements](#_heading=h.jzks6944xfem) 20

[5.4.5.Jobs](#_heading=h.8u50yje54i2s) 20

[5.4.6. Others](#_heading=h.2w5ecyt) 20

[5.5. Configuration](#_heading=h.3vac5uf) 20

[5.5.1. Operating System](#_heading=h.2afmg28) 20

[5.5.2. Database](#_heading=h.pkwqa1) 20

[5.5.3. Desktop](#_heading=h.1opuj5n) 20

[**6.REFERENCES**](#_heading=h.agy4l6qgni4e) **20**

[**7. APPENDIX**](#_heading=h.2250f4o) **21**

# 1.INTRODUCTION

This project is intended to implement a feature that allows you to send all the calls that come to your mobile to another mobile of your choosing. Selective Call Forwarding allows you to send selected incoming calls to a different number. Call forwarding is a network setting and happens in less than a second. Unless your network has some exotic setting that informs you, you will not know whether your call is forwarded or not.

## 1.1.Intended Audience

This document could be shared or viewed across all the following CG employees,BU SMEs, internal SMEs. This is a technical document, and the terms should be understood by all of them.

| CG Employees |  |
| --- | --- |
| BU SMEs |  |
| Internal SMEs |  |

## 1.2.Acronyms/Abbreviations

| NA | Not Applicable |
| --- | --- |
| CFSS | CALL FORWARDING SYSTEM SIMULATOR |
| CG | Capgemini |
| SME | Small and Medium sized Enterprises |
| BU | Business Unit |

## 1.3.Purpose of the project

The purpose of this project is it helps you to redirect or forward incoming calls to an alternate number. It is commonly used to forward calls to an office phone to a user's cell or home phone, or a colleague's number.

## 1.4.Key project Objectives

* Always Availability and Better Business Presence
* Multiple Extensions on a single number
* Better Customer-Agent Interactions

## 1.5.Project scope and limitation

### 1.5.1.In scope

The scope of the project is limited to building a call forwarding simulator application consisting of a configurable database and users.

### 1.5.2.Out of Scope

It is not a real time project but it is a simulation.

## 1.6.Functional Overview

* Call Forwarding Service Simulator works in a user - client model. If someone is out of the office for vacation or work travel, incoming calls can be sent to an alternate number, ensuring that callers get questions answered and issues resolved.So all forwarding requests made by cell phone are saved and served by the server in the database. So all the forward related settings are maintained at network provider level in a database. CFSS supports 3 types of call forwarding services i.e unconditional or No reply or as Busy
* Unconditional : All calls will be forwarded to that particular person.
* No reply : If the call rings for a particular timestamp and the person didn’t pick the call, then the call will be forwarded.
* Busy : The person can forward the call if he/she is busy.

## 1.7.Assumptions, Dependencies & Constraints

Issues facing if any to avoid we need a unix like environment to perform the task and reliable connectivity while file searching.

## 1.8.Risks

NA. There are no risks for Call Forwarding System Simulator.

# 2.DESIGN OVERVIEW

## 2.1.Design Objectives

The goal of call forwarding is to redirect or forward incoming calls to an alternate number.

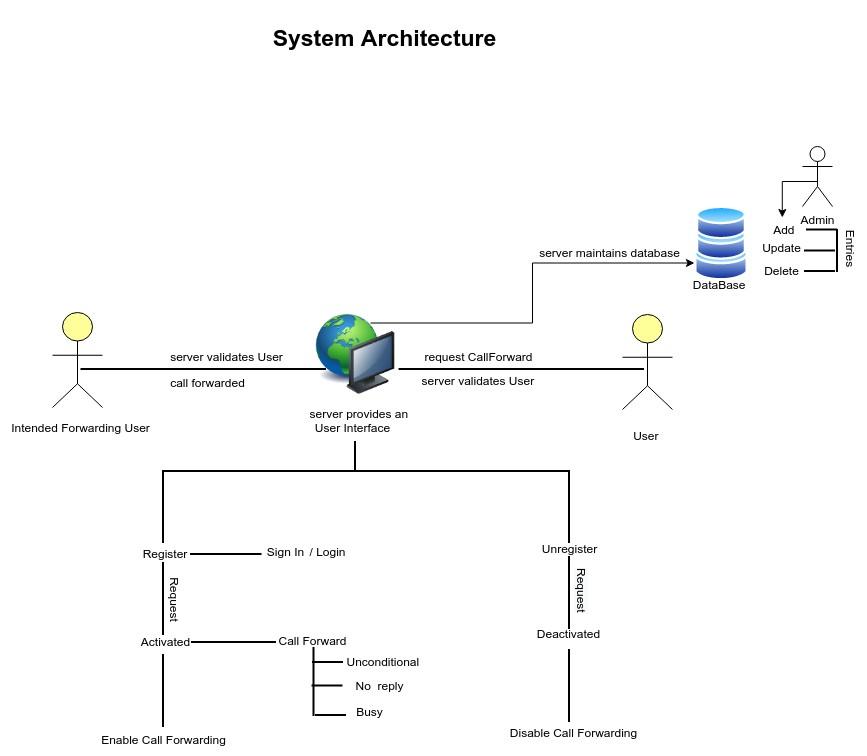
### 

### 

### 

### 2.1.1.Recommended Architecture

The recommended system architecture is as follows.



## 2.2.Architectural Strategies

### 2.2.1.Design Alternative

The project uses a database to establish a connection between the user and server for call forwarding services.

### 2.2.2.Reuse of Existing Common Services/Utilities

The project does not reuse any new common services or utilities.

### 2.2.3.Creation of New Common Services/Utilities

The project does not create or use any new common services or utilities.

### 2.2.4.User Interface Paradigms

* Desktop or a Linux machine with internet connection.
* Command Line Interface (CLI).

### 2.2.5.System Interface Paradigms

* Operating system – Unix.
* Linux Kernel version - 4.4.0-19041-Microsoft.
* Bash shell: x86\_64 GNU/Linux

### 2.2.6.Error Detection / Exception Handling

* Error detection in all phases of client connection to the server will be provided.
* Four levels of debug log messages will be included like FATA, INFO, WARNING & DEBUG.
* Appropriate error message for file handling will also be included

### 2.2.7.Memory Management

NA

### 2.2.8.Performance

* Quick response
* System design will fulfill performance requirements.

### 2.2.9.Security

Any anonymous user tries to access the server; it allows the corresponding user to access for read only mode.

### 2.2.10.Concurrency and Synchronization

NA

### 2.2.11.Housekeeping and Maintenance

* Clearing the memory buffers from the system.
* Flushing the contents of the screen when the User starts for better Experience.

# 3.SYSTEM ARCHITECTURE

Call Forwarding System is a networking connection between user and server.The user connects with the server through a software.

## 3.1.System Architecture Diagram

NA

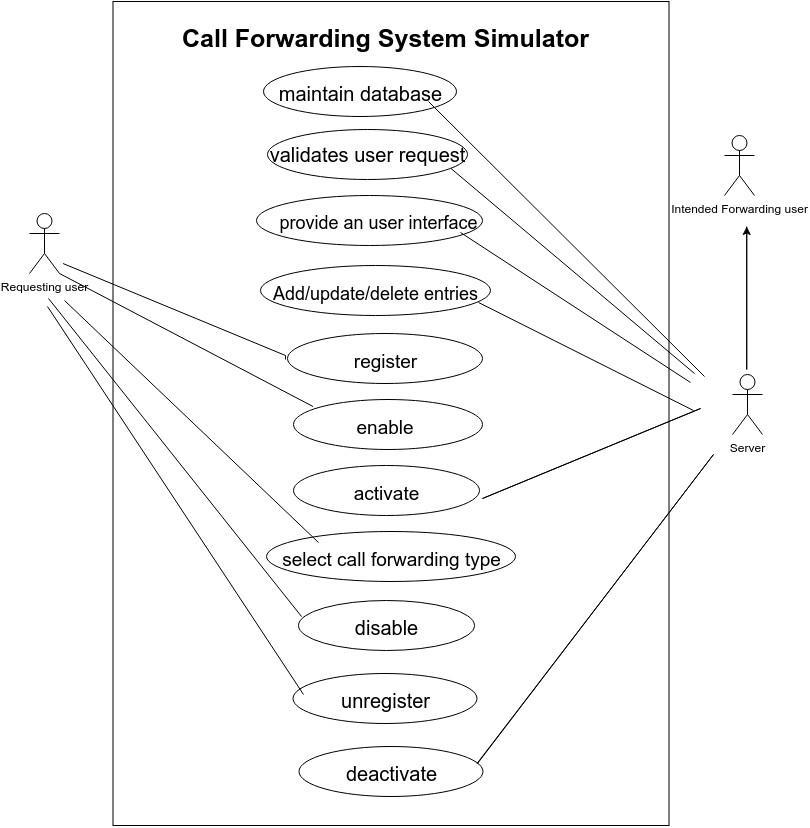
## 3.2.System Use-Cases

* Users register and sign in into the application.
* The application is activated when detected busy , no reply, unconditional

call is auto forwarded to the intended forward user.

* When deactivated, the call forwarding is disabled.

## 3.3. Subsystem Architecture



## 3.4. System Interface

### 3.4.1.Internal Interfaces

The internal interfaces comprise interfaces through which the system interacts with the user through which it provides them services.

* Cloud System
* Internet

The internal interfaces comprises interfaces through which the system

* Interacts with the clients through which it provides them services.

1. Cloud System
2. Internet

### 3.4.2. External Interfaces

The external interface comprises interfaces through which the users interact with the system.

* Desktop or Linux Machine
* Internet
* Software where the application is activated.

# 4.DETAILED SYSTEM DESIGN

The Call Forwarding application basically includes a server and user connection.

The user can perform the forwarding application: -

1.By establishing connection with the indented forward user

## 4.1.Key Entities

The key entities associated with the system are

Server

* The server is a remote entity that maintains the database.
* It provides entries to Add , Update ,and Delete any activity.
* Provides service on the user request.
* It provides Server Interface.

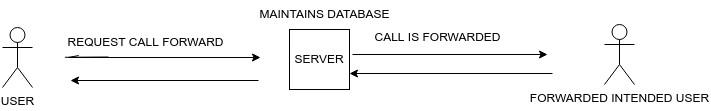
User

* User is an entity which requests to connect with the server.
* User requests the server to enable the call forwarding application by registering into the application and by signing in.

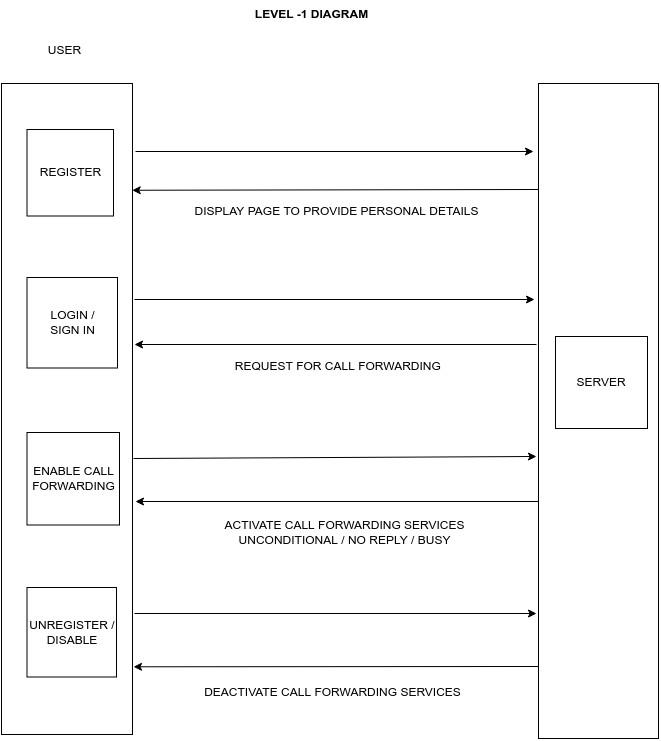
## 4.2.Detailed-Level Database Design

### 4.2.1.Flowchart

### 4.2.2 Level- 0 Diagram

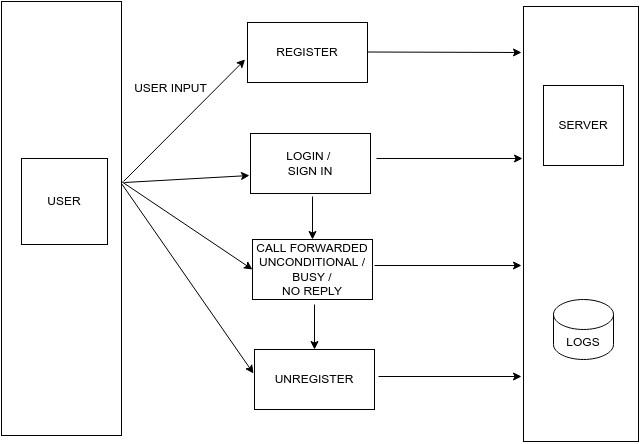


### 4.2.3 Level -1 Diagram

****

### 4.2.4 Level -2 Diagram



****

## 4.3.Requirements Specifications

### 4.3.1.Register and Unregister

The system shall allow new user to register into the application and sign in

according to the given credentials and should also allow existing user to

unregister from that application according to their requirement.

### 4.3.2.Support three types of call forwarding

The system shall support three types of services that are unconditional , busy and no reply.

### 4.3.3.Support normal call

The system shall support normal calls when the call forwarding service is deactivated.

### 4.3.4.Archival and retention requirements

Logs have been maintained on the server side, according to what function server is performing.

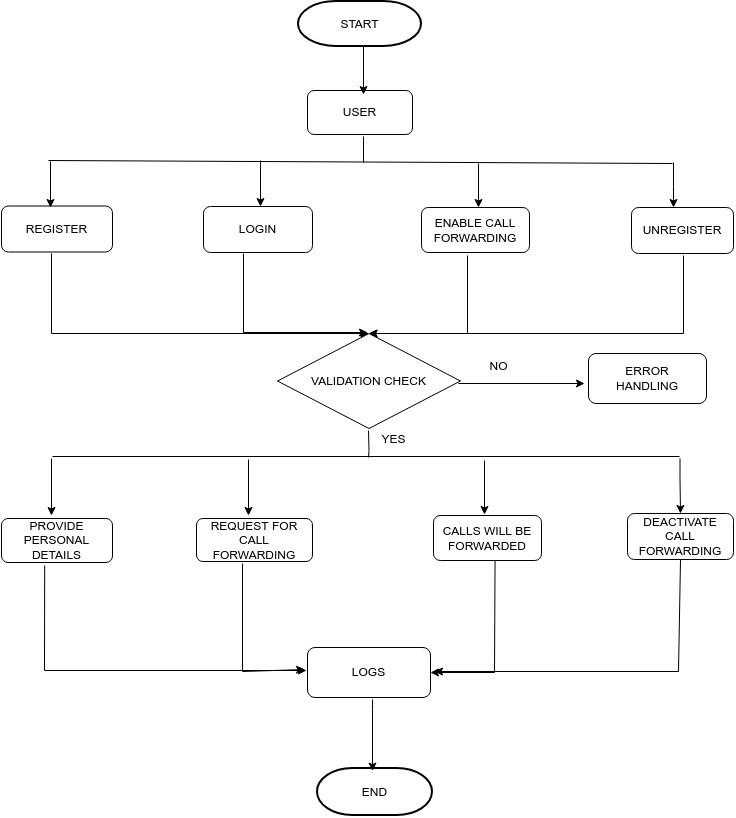
RSE includes debug log message with at least 4 levels

FATA, INFO, WARNING, DEBUG.

## 4.4. Disaster and Failure Recovery

NA.

## 4.5.Business Process Workflow



## 4.6.Variables

struct userPersonalDetails - id , phNo,uName ,gender,regFlag

userPersonalDetails \*next

struct loginDetails - id , uName , passwd,loginDetails \*next

struct callForwarding - id,regFlag,cfsNumber,cfsActive,status

callForwarding \*next

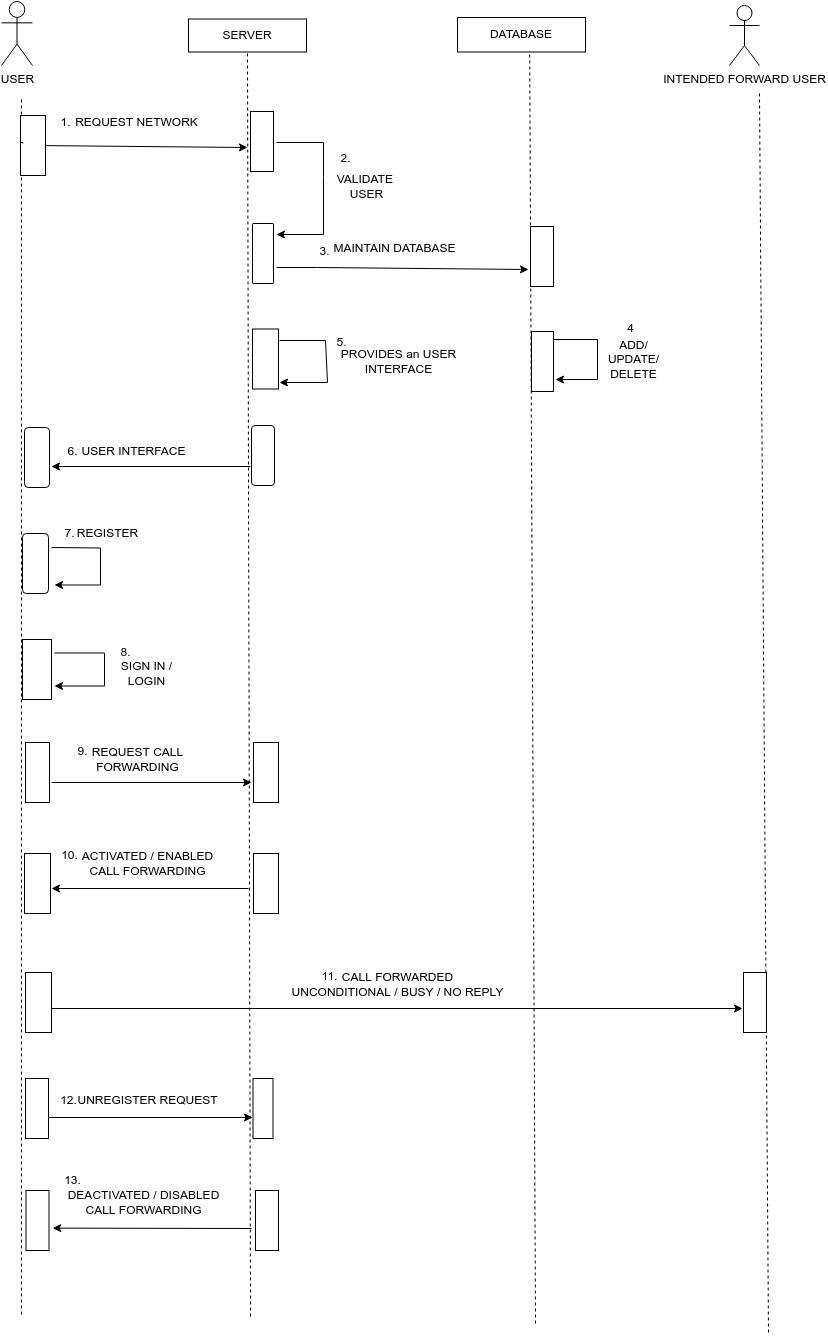
## 

## 

## 

## 4.7.Activity Diagram

### 4.7.1.Sequence Diagram

****

## 4.9. Data Migration

Data is migrated between the user and the server.

# 5.ENVIRONMENT DESCRIPTION

## 5.1.Time Zone

It will support time zones as per Indian standard time (IST) in (GMT +5:30) and UST standard.

## 5.2.Language Support

C language and compilation using make file and gcc. The Linux commands to do that task we can specify the commands.

## 5.3.User Desktop Requirements

User desktop requires a Linux environment, Operating system of Linux Debian or Terminal x86\_64 GNU/Linux and kernel version 4.4.0-19041-Microsoft #1237 Microsoft and reliable internet connectivity.

## 5.4.Server-Side Requirements

In server side,

● Disk space – Minimum 150GB

● Uninterrupted connectivity 24x7

● Monitor long running jobs, to reduce the server load

### 5.4.1.Deployment Considerations

Deployment considerations are,

● 500Mhz Processor

● 120GB HDD CPU

● minimum 4GB RAM

● Network connectivity

### 5.4.2. Application Server Disk Space

Disk space – Minimum 150GB

### 5.4.3.Database Server Disk Space

NA

### 

### 5.4.4.Integration Requirements

The PWD Displays the current working directory on the server for the logged in user

### 5.4.5.Jobs

When someone activates call forwarding on their phones, it means that they want their incoming calls on that particular number to be redirected to another phone number of their choice. It is especially useful in instances of network problems or other reasons. In business ,the human touch can improve contact, thus sales but traditional wired answering services are expensive, so they can have their call forwarded to a call center, so that clients can reach an operator ,and they can get their problem solution easily.

### 5.4.6. Others

NA.

## 5.5. Configuration

* Call forwarding system helps us to manage our inbound call flow, depending on whether the user is busy or not available to answer an incoming call.
* Call forwarding Always: Here all calls are forwarded.
* Call forwarding Busy:Calls are forwarded when the user is busy with another phone.
* Call forwarding no answer: here calls are forwarded when users do not answer the phone .

### 5.5.1. Operating System

* Operating system –Linux.
* RAM - 4GB or more.
* Processor - i3/i5.

### 5.5.2. Database

NA

### 5.5.3. Desktop

Minimum Windows 10, 4 gb Ram with i3 configuration is required.

# 

# 6.REFERENCES

<https://novocall.co/blog/what-is-call-forwarding/> About Call Forwarding

<https://www.javatpoint.com/linux-commands> Linux Command

<https://www.javatpoint.com/file-handling-in-c> File Handling Concept

<https://stackoverflow.com/questions/2031163/when-to-use-thedifferentloglevels>

And The documents And pdf which Capgemini provided us.

# 7. APPENDIX

NA

**Change Log**

| **CFSS Template Version Control (Maintained by QA)** | | | | | |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |
| **Date** | **Version** | **Author** | | **Description** | |
| 13-Nov-2022 | 0.1 | Group04 | | Initial Version | |
|  |  |  | |  | |
|  |  |  | |  | |
|  |  |  | |  | |