DND

0.1V

DO NOT DISTURB (DND)

The purpose of this document is to provide with a template for documenting DND

**Document Control :**

| **Project Revision History** | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  | |  |  |  |  | |  |
| **Date** | **Version** | **Author** | **Brief Description of Changes** | | | | **Approver Signature** |
| 18/11/22 | 0.1 | Group 03 | Initial Draft | | | |  |
| 21/11/22 | 0.2 | Group03 | Initial Draft | | | |  |
| 25/11/22 | 0.3 | Group03 | Finalize the Documentation | | | |  |

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

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

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

[1.3. Project Purpose 5](#_heading=h.3dy6vkm)

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

[1.5. Project Scope and Limitation 6](#_heading=h.4d34og8)

[1.5.1. In Scope 6](#_heading=h.4d34og8)

[1.5.2. Out of scope](#_heading=h.2s8eyo1)6

[1.6. Functional Overview 6](#_heading=h.17dp8vu)

[1.7. Benefits of dnd](#_heading=h.4h042r0)

[1.8. Risks 6](#_heading=h.3rdcrjn)

[**2. Design Overview 6**](#_heading=h.26in1rg)

[2.1. Design Objectives 6](#_heading=h.lnxbz9)

[2.1.1. Recommended Architecture](#_heading=h.35nkun2)6

[2.2. Architectural Strategies 6](#_heading=h.1ksv4uv)

[2.2.1. Design Alternative 7](#_heading=h.44sinio)

[2.2.2. Reuse of Existing Common Services/Utilities 7](#_heading=h.2jxsxqh)

[2.2.3. Creation of New Common Services/Utilities 7](#_heading=h.z337ya)

[2.2.4. User Interface Paradigms 7](#_heading=h.3j2qqm3)

[2.2.5. System Interface Paradigms 7](#_heading=h.4i7ojhp)

[2.2.6. Error Detection / Exceptional Handling 7](#_heading=h.2xcytpi)

[2.2.7. Memory Management 7](#_heading=h.1ci93xb)

[2.2.8. Performance 7](#_heading=h.3whwml4)

[2.2.9. Security 7](#_heading=h.qsh70q)

[2.2.10. Concurrency and Synchronization 8](#_heading=h.qsh70q)

[2.2.11. Housekeeping and Maintenance](#_heading=h.1pxezwc)8

[**3. System Architecture 8**](#_heading=h.49x2ik5)

[3.1. System Architecture Diagram. (Not Necessary) 8](#_heading=h.147n2zr)

[3.2. System Use-Cases 8](#_heading=h.3o7alnk)

[3.3. Subsystem Architecture 8](#_heading=h.ihv636)

[3.4. System Interfaces 8](#_heading=h.32hioqz)

[3.4.1. Internal Interfaces 9](#_heading=h.1hmsyys)

[3.4.2. External Interfaces](#_heading=h.41mghml)9

[**4. Detailed System Design 9**](#_heading=h.2grqrue)

[4.1. Key Entities 9](#_heading=h.vx1227)

[4.2. Detailed-Level Database Design 9](#_heading=h.3fwokq0)

[4.2.1. Data Mapping Information 9](#_heading=h.1v1yuxt)

[4.2.2. Data Conversion](#_heading=h.4f1mdlm)10

[4.3. Archival and retention requirements 10](#_heading=h.2u6wntf)

[4.4. Disaster and Failure Recovery 10](#_heading=h.19c6y18)

[4.5. Business Process workflow 10](#_heading=h.3tbugp1)

[4.6. Business Process Modeling and Management (as applicable) 10](#_heading=h.28h4qwu)

[4.7. Business Logic 10](#_heading=h.nmf14n)

[4.8. Variables 10](#_heading=h.37m2jsg)

[4.9. Activity / Class Diagrams (as applicable) 10](#_heading=h.1mrcu09)

[4.10. Data Migration 10](#_heading=h.46r0co2)

[4.10.1. Architectural Representation 10](#_heading=h.2lwamvv)

[4.10.2. Architectural Goals and Constraints 10](#_heading=h.111kx3o)

[4.10.3. Logical View 11](#_heading=h.3l18frh)

[4.10.4. Architecturally Significant Design Packages 11](#_heading=h.206ipza)

[4.10.5. Data model](#_heading=h.4k668n3)11

[4.10.6. Deployment View](#_heading=h.2zbgiuw)11

[**5. Environment Description 11**](#_heading=h.1egqt2p)

[5.1. Time Zone Support 11](#_heading=h.3ygebqi)

[5.2. Language Support 11](#_heading=h.2dlolyb)

[5.3. User Desktop Requirements 11](#_heading=h.sqyw64)

[5.4. Server-Side Requirements 11](#_heading=h.3cqmetx)

[5.4.1. Deployment Considerations 12](#_heading=h.1rvwp1q)

[5.4.2. Application Server Disk Space](#_heading=h.4bvk7pj)12

[*5.4.3.* Database Server Disk Space](#_heading=h.2r0uhxc)12

[5.4.4. Integration Requirements 12](#_heading=h.1664s55)

[5.4.5. Jobs 12](#_heading=h.3q5sasy)

[5.4.6. Network 12](#_heading=h.25b2l0r)

[5.4.7. Others](#_heading=h.kgcv8k)12

[5.5. Configuration 12](#_heading=h.34g0dwd)

[5.5.1. Operating System 12](#_heading=h.1jlao46)

[5.5.2. Database 12](#_heading=h.43ky6rz)

[5.5.3. Network 12](#_heading=h.2iq8gzs)

[5.5.4. Desktop](#_heading=h.xvir7l)12

[**6. References 13**](#_heading=h.3hv69ve)

[**7. Appendix 13**](#_heading=h.1x0gk37)

### 

# 1.Introduction

This project is intended to implement a feature that blocks all the calls that come to your mobile. Selective DND mode allows you to block selective calls to a different number. With this feature, it's easier to mute/block all calls and alerts for tools like mobile /Androids.

## 1.1. Intended Audience

This document could be shared or viewed across all the following members :

CG employees, BU SME's , internal SME's.

This is a technical document, and the terms should be understood by all of them.

| CG Employee |  |
| --- | --- |
| BU SME’ S |  |
| Internal SME’S |  |

## 1.2.Acronyms/Abbreviations

| DND | DO NOT DISTURB |
| --- | --- |

## 1.3.Project Purpose

The Do not disturb mode on your mobile phone is used to block the incoming Calls. The purpose of the Project is to implement the do not disturb feature which allows you to mute or block the incoming call alerts. With this feature, it's easier to mute/block all calls and alerts for tools like mobile /Android.

## 1.4.Key Project Objectives

* The Do not disturb mode on your mobile phone is used to block the incoming Calls.

## 1.5.Project Scope and Limitation

## 1.5.1 In Scope

The purpose of this application is to implement a user of real-time communication that does not wish to be interrupted by incoming calls.

### 1.5.2 Out of scope

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

## 1.6.Functional Overview

Do Not Disturb mode works in a User – client model. If someone is busy or in an important meeting, then incoming calls can be blocked by the User. The incoming calls can be blocked by the user in two ways i.e, either by Selectively mode or Globally mode.

1.Selectively Mode: In Selectively Mode ,we can select few of the contact numbers and the respective incoming calls will be blocked.

2.Globally Mode: In Globally mode all the incoming calls will be blocked by the user.

## 1.7.Benefits of DND

1. If someone is in the meeting or driving, they don’t want to get interrupted in the middle. so, incoming calls can be blocked by keeping their mobile in the DND mode.
2. So all the calls that the user got while the mobile is on DND mode are saved and served by the server in the database.
3. Allows you to register your DND preferences as to what kind of numbers you want to block or whether you want to block all kinds of commercial communication.

## 1.8. Risks

In case, if we have any kind of emergency, we can’t reach that person who has enabled DND mode on their mobile. They can not get any kind of message.

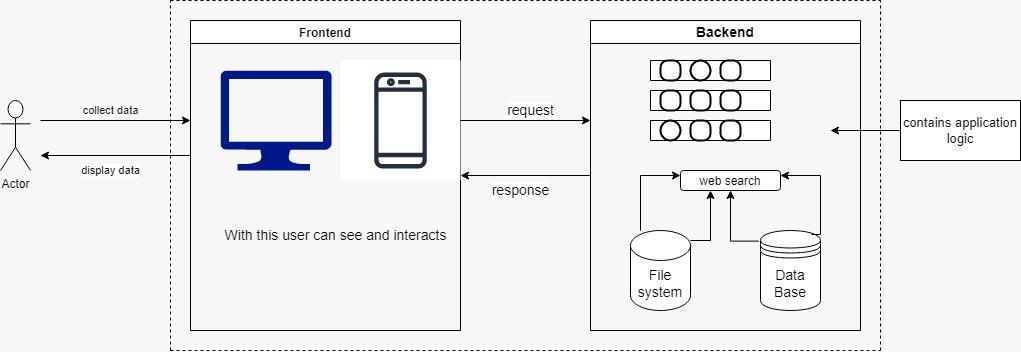
# 2. Design Overview

## 2.1. Design Objectives

The goal of DND is to block the incoming call.

### 2.1.1 Recommended Architecture

The recommended system architecture is as follows.



## Fig 1

## 2.2.Architectural Strategies

### 2.2.1 Design Alternative

The project uses a database to establish a connection between the user and server for DND 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 / Exceptional 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 messages 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

Anonymous users cannot access the data.

### 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 user starts for better Experience.

# 3.Experience System Architecture

DND is a networking connection between user and server. The user connects with the server through software.

## System Architecture Diagram. (Not Necessary)

NA

## System Use-Cases

## Users register and sign in into the application.

## The application is activated then detected as busy,or off. Call automatically gets out of the network.

* When deactivated, the call DND mode is disabled

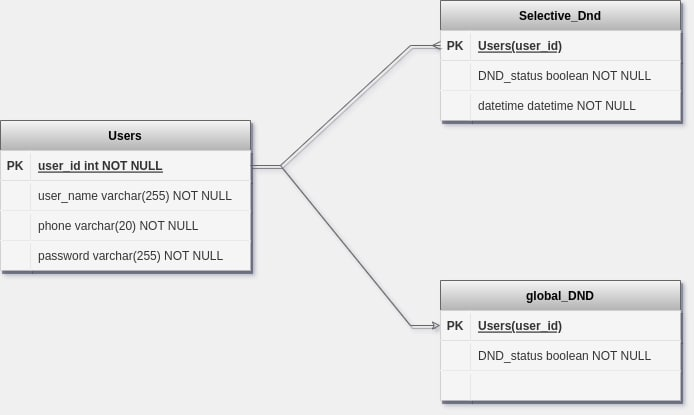


Fig 2

## Subsystem Architecture

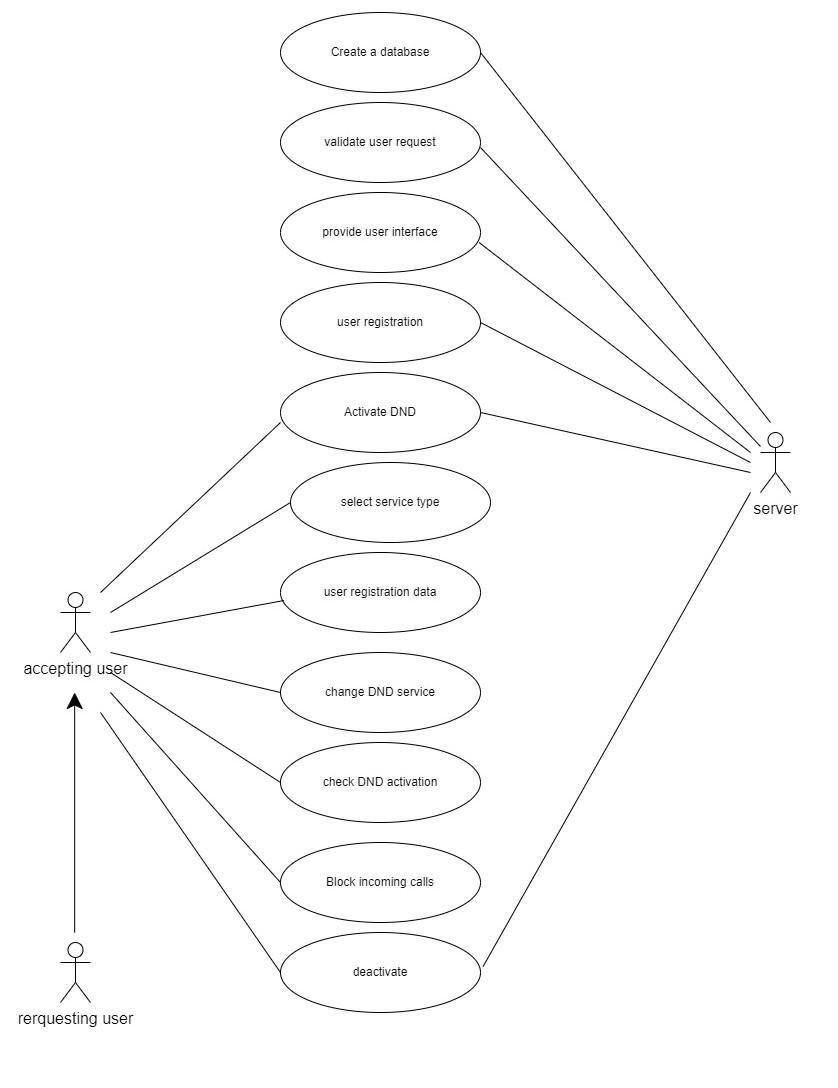


Fig 3

## System Interfaces

## 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 comprise 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 DND application basically includes a server and user connection.

The user can perform the DND application: -

1.By enabling DND, users can inactive all the applications.

## 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 DND, global ,active, inactive
* Provides service on the user request.
* It provides Server Interface.

**User**

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

## Detailed-Level Database Design

**4.2.1.1 Level - 0 Diagram**



Fig 4

**4.2.1.2 Level - 1 Diagram**

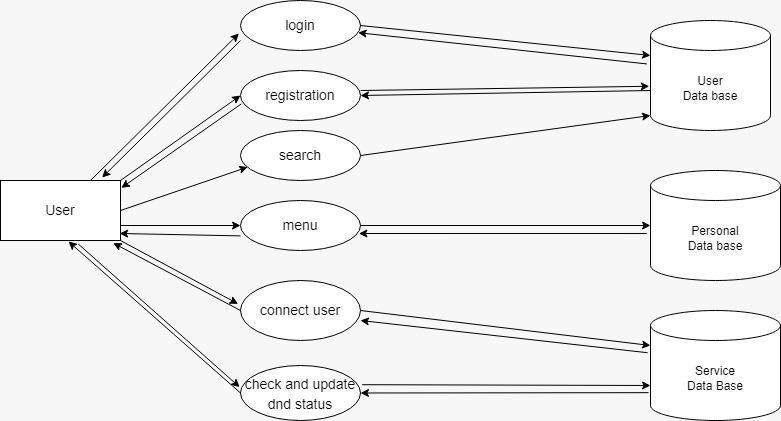


Fig 5

### 4.2.1 Data Mapping Information

NA

### 4.2.2 Data Conversion

NA

## Archival and retention requirements

The system shall allow new users to register into the application and sign in according to the given credentials and should also allow existing users to unregister from that application according to their requirement.

## Disaster and Failure Recovery

NA

## Business Process workflow

## NA

## Business Process Modeling and Management (as applicable)

NA

## Business Logic

NA

## 4.8 Variables

NA

## 4.9Activity / Class Diagrams (as applicable)

NA

## 4.10Data Migration

Data is migrated between the user and the server.

### 4.10.1 Architectural Representation

NA

### 4.10.2 Architectural Goals and Constraints

NA

### 4.10.3 Logical View

NA

### 4.10.4 Architecturally Significant Design Packages

NA

### 4.10.5 Data model

NA

### 4.10.6 Deployment View

NA

# 5.Environment Description

The environment description allows the user to block the calls remotely by selecting options like selectively or globally based on the user requirements.

## 5.1 Time Zone Support

It will support the time zone as per Indian Standard Time (IST)in(GMT +5:30) and UST standard

## Language Support

C language and compilation using 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 Ubuntu 20.04.5 LTS (GNU/Linux 4.4.0-19041-Microsoft x86\_64) kernel version 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

● 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 Requirement

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

### 5.4.5 Jobs

NA

### 5.4.6 Network

The network connects the system for the purpose of file searching therefore stable Internet Connectivity is required

### 5.4.7 Others

NA

## Configuration

NA

### 5.5.1 Operating System

* Operating system – Unix.
* RAM - 8GB.
* Processor - i3/i5/i7.

### 5.5.2 Database

NA

### 5.5.3 Network

The following are the network details regarding the project:

* + The client and server communicate over a UDP/IP protocol.
  + The IP address used can be either IPv4 or IPv6.

### 5.5.4 Desktop

Unix like environment is required

# 6. References

1. <https://www.codespeedy.com/find-a-specific-file-in-a-directory-in-cpp/> Find a Specific File in a directory.
2. System Requirements Specification Document
3. <https://linuxhint.com/grep_command_linux/> Grep Command in Linux

### 7.Appendix

NA