

S O F T W A R E
R E Q U I R E M E N T
S P E C I F I C A T I O N

COURSE PROJECT ~
MiniBuzz

Course Project Partners ~

Neha Soni (B19CSE058)

Nikhil Raj (B19CSE059)

❑ P R E F A C E ~

The most Prominent Motivation behind the Project (MiniBuzz) is to provide the users with a 'Not So Boring' kind of Interactive Interface, where Users can Post Live Queries & Interact To other users Posts' by Replying to their Queries.

This Document States & Describes the Problem Statement that MiniBuzz intends to Address, its Scope & Context, all the Technical Functional & Non-Functional Requirements from User Perspective along with Other Requirements & Constraints.

It is Intended for All the Stakeholders Involved in the Project to Aid in Process Development & to Precisely Serve as a Reference for Clarification & Future Assistance.

○ PROJECT OUTLINE ~

- ❑ The Project brushes up with the Problems Faced in Resolving Aroused Queries of the Students.
- ❑ Provides an Interacting, Easy & User Friendly Interface to Interact with other People.
- ❑ Resolve Queries.

○ PROJECT DOMAIN ~

- ❑ It Addresses to the Queries, Students are Facing.
- ❑ Resolves Queries by Providing Other Users to React on Your Query.

○ SRS DOCUMENT USERS ~

- | | | |
|----------------------|----------------------|--------------------------|
| ❑ IT TEAM | ❑ DEVELOPERS | ❑ PROGRAMMERS |
| ❑ PROJECT
MANAGER | ❑ FINANCE
MANAGER | ❑ SOFTWARE
ARCHITECTS |
| ❑ DESIGNERS | ❑ TESTING TEAM | ❑ DEVELOPMENT
TEAM |

❑ INTRODUCTION ~

The Document cover all aspect's that'd be required in the complete fulfilment of the product, its relevant context, Problem specification, scope of the document & Structure of the Document :

○ CONTEXT

- ☐ The Application Solely Encompasses Student's Fraternity.
- ☐ Designing Query Raising Application With Server & Users, to Raise up all Doubts & Questions (POST), Over any Range of Topic
- ☐ Student's could Get Responses From Other Users (COMMENT)
- ☐ Allows Users, With Intent, to Add up BIO (Short Description)
- ☐ Resolving Student's Doubts & Queries is the Application-Software's Prime point of Concern
- ☐ Users could also Contact other Users For Further Clarification by looking their ID'd or Details Provided
- ☐ Any User With Minimal Knowledge of Operating App's Will be able To Use the Software Application

○ PROBLEM SPECIFICATION

- ☐ Not Having a platform Precisely Specific to Students Queries
- ☐ Providing Users a platform to interact & comment on each others raised Queries
- ☐ Providing Users a Means to Get in Contact With the Other User Whom they wish to get in touch with
- ☐ Providing User Friendly & easy to use Environment throughout

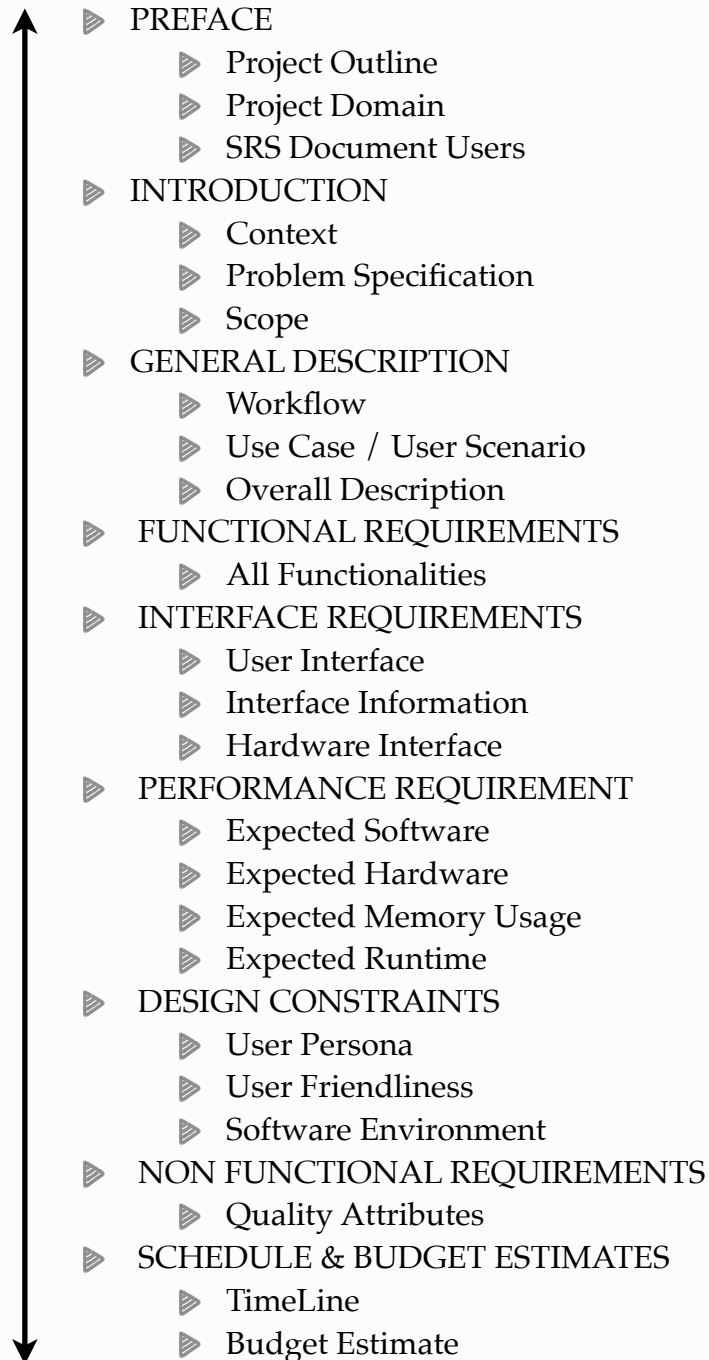
○ SCOPE

- ☐ It Encompass Wide Range of User Domain (Students With Queries)
- ☐ Covers up all functionalities required by the user, considering the use case scenarios from end user point of view
- ☐ Covers All Non functional requirements (Quality Attributes) , that'd be implemented with the project
- ☐ The Interface required while using the Application
- ☐ Keeps an Eye on Hardware Constraints, Performance Expected
- ☐ Covers the Schedule timeline & budget Estimates that'd be required to complete the project From initial Requirement Gathering to Deployment of product

If Any Requirement Comes Out of Context, This Document will NOT Handle It !

○ OVERVIEW & STRUCTURE

□ Document Structure ~



□ Document Overview ~

- ▶ **GENAERAL DESCRIPTION ~**
 - ▶ Workflow ~ Transition between various Interfaces
 - ▶ All Use Cases & User Scenarios as per End User Perspective
- ▶ **FUNCTIONAL REQUIREMENTS ~**
 - ▶ All Functionalities as per Requirements of User
 - ▶ Detail of Individual Functionalities

▶ **INTERFACE REQUIREMENTS ~**

- ▶ Detail of All Interfaces, Orientation, Controls & Buttons that user confronts along with Information Displayed
- ▶ Hardware Interface Details

▶ **PERFORMANCE REQUIREMENT ~**

- ▶ Detail of Expected Software , Hardware , Performance, Runtime & Memory Usage as Per User Demand
- ▶ Operating Environment attributes

▶ **DESIGN CONSTRAINTS ~**

- ▶ Detail Regarding User Personas ~ all different users of the software application
- ▶ Information on How Friendly is the software application
- ▶ Details of Software Environment Compatibility

▶ **NON FUNCTIONAL REQUIREMENTS ~**

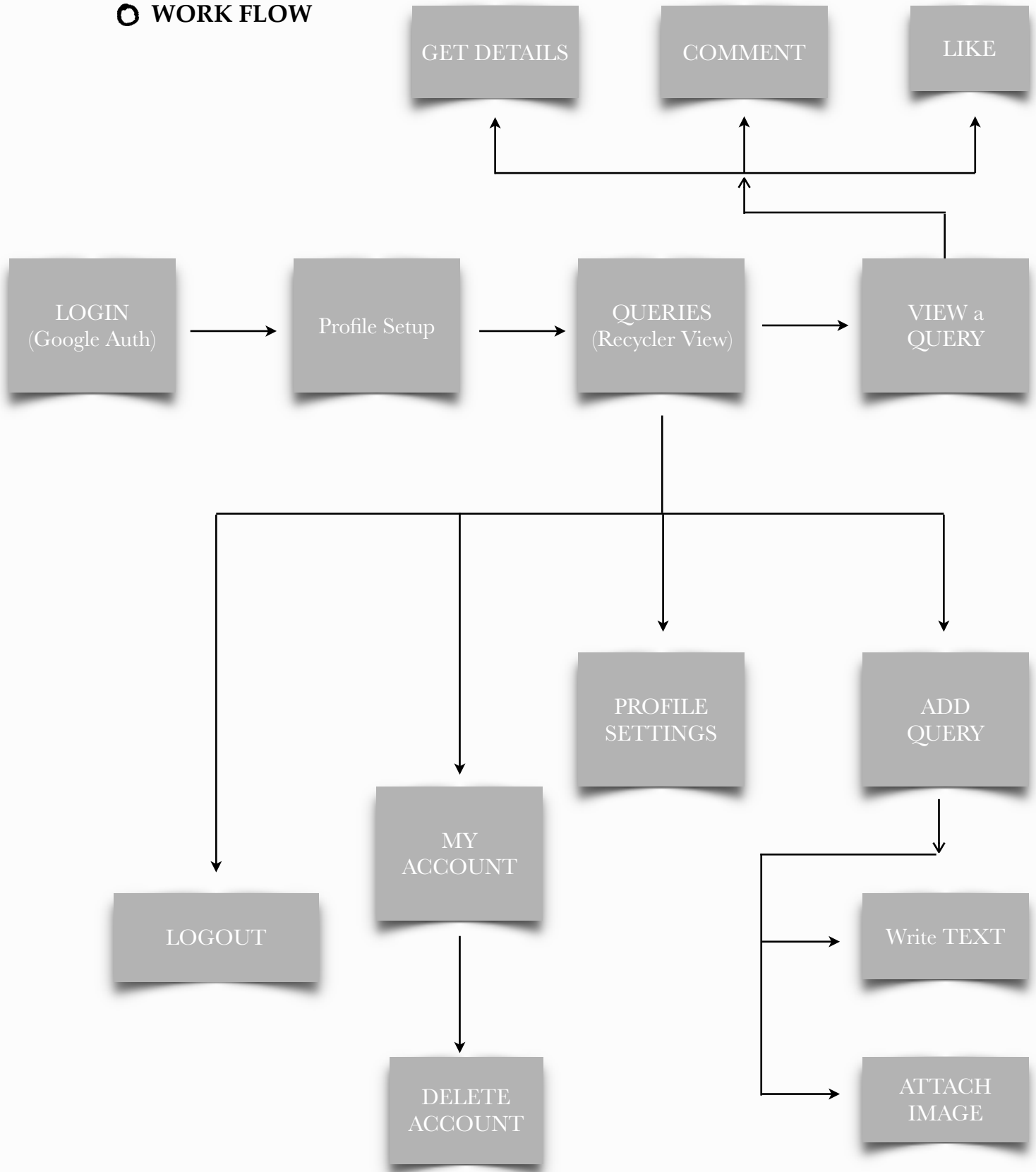
- ▶ Functional Suitability
- ▶ User Interface Aesthetics
- ▶ Learnability

▶ **SCHEDULE & BUDGET ESTIMATES ~**

- ▶ Mention of Timeline of Project
- ▶ Budget Estimate Calculation for the Proposed Timeline

□ GENERAL DESCRIPTION ~

○ WORK FLOW



○ USE CASE / USER SCENARIO

- ☐ **Login**
 - ▶ Logging In Uniquely & Securely into the System
- ☐ **Profile Setup**
 - ▶ Setting Up User Profile (Name / Details / User Bio)
- ☐ **View My Account**
 - ▶ Viewing User's Own Account
- ☐ **Profile Settings**
 - ▶ Changing Profile Settings (Name / Details / User Bio)
- ☐ **Add a Post**
 - ▶ Raising / Posting Queries Online (Text / Image)
- ☐ **View Queries**
 - ▶ Scrolling Through the Posts & Queries of Other Users
- ☐ **View Profiles**
 - ▶ Viewing Other User's Profile's
- ☐ **Interact**
 - ▶ Reacting on Other User's Queries & Responding to it
 - ▶ Like
 - ▶ Comment
- ☐ **Logout**
 - ▶ Successfully Logging Out of The System
- ☐ **Delete Account**
 - ▶ Deleting Account When Needed

○ SOFTWARE PRODUCT ~ OVERALL DESCRIPTION

- ☐ Sole Perspective & Objective of MiniBuzz is Though End User Point of View,
- ☐ Provides a User Friendly Environment, Where Queries Could be Resolved Directly, Without being Digressed from the Topic
- ☐ It's a Utility Software Designed Specifically for Students Fraternity.
- ☐ It's an Android (Touch Gesture) Based Application
- ☐ It Authorise Users to Uniquely Login into the system with their G-Mail ID, Logout on their demand & Delete their Account.
- ☐ Software lets the Users (Students) Interact on a common Query Handle, Where they can Raise a Query (Post) & Respond to an already Posted Query (Comment).
- ☐ Users can Review Previous Posts & Other Users Profile too
- ☐ It can be Moderately Personalised ~
 - ▶ Users Can Upload Their Profile Picture
 - ▶ Choose a Profile Name
 - ▶ Write Up a Bio as per Users Wish

❑ FUNCTIONAL REQUIREMENTS ~

Considering All Use Case Scenarios and, Viewing in Mind End - User Perspective,
Explicit Functional Requirements ~

○ LOGIN

Allow users to Log in Uniquely into their Specific Accounts.

- ☐ Login With Google Authentication
- ☐ Authorised User is Registered in Firebase Authentication List
- ☐ Authorised User will be Registered on the Firebase Database
- ☐ Changing Password & Reset Password Will Be Achieved Through Google's Side.

○ PROFILE SETUP

User Can Set Up the Profile After Being Logged in Into the System.
Set Up ~

- ☐ Adding Full Name
- ☐ Adding Other Applications' Profiles (e.g. Contact Number , Insta Account , Git Hub ID etc)
- ☐ Writing up a BIO (Short Description Provided by the USER)

○ MY @ ACCOUNT

User can View his/her Own Account, consisting of the following ~

- ☐ Profile Image
- ☐ Other Added Profile ID's
- ☐ Bio (Description Provided)
- ☐ Previous Posts

○ PROFILE SETTINGS

Allow User To Change the Prior Provided Data.

- ☐ User Full Name
- ☐ Other Account's Provided
- ☐ Bio (Description)
- ☐ Upload User Profile Image

○ POST @ QUERIES

- ☐ Allow User To Post Queries / Questions
- ☐ Allow User To Upload Relevant Images in context with the Query (Text)

○ QUERIES SECTION - RECYCLER VIEW

- ☐ Allow User To View Up all Other Queries raised by various Other Users, in Recycler View Layout
- ☐ User Can Click on any Query To Comment on It
- ☐ User can Click on an Profile pic on Query to View that User Profile

○ COMMENT / UNCOMMENT

- ☐ Allow User To React To Other Raised Queries by Responding to it
- ☐ User Can Comment Text
- ☐ User can Upload Relevant Images in context of that Query
- ☐ Allow user to Delete their Comment

○ LIKE / UNLIKE

- ☐ User Will be receiving Notifications When someone Comment on their Poster / Raised Query.
- ☐ Allow user to unlike a post they once liked

○ LOGOUT

- ☐ User can Successfully Logout

○ DELETE ACCOUNT

- ☐ User can Delete the Account Successfully, If Wished

❑ INTERFACE REQUIREMENTS ~

○ USER INTERFACE

- ☐ It's A Mobile Android based Application, with Touch Gestures
- ☐ Easy to handle software
- ☐ Anyone (STUDENT) with basic App handling knowledge will be able to get on with the interface

Various Interfaces :

- ☐ Login/Logout interface
- ☐ Queries Section interface (recycler View)
- ☐ Profile Interface
- ☐ Account interface
- ☐ Comment Interface
- ☐ Adding Post Interface

○ INFORMATION DISPLAY

- ☐ Information Displayed on the Screen Depends on various Interfaces
- ☐ Login page shows up Google login option
- ☐ Queries Section shows already posted Queries, With Scroll view in recycler View, It also contain user profile image at the top.
- ☐ Profile page contains text input for name , other accounts , small Description text box
- ☐ Account page contain user profile data, along with users prior posted Queries

○ HARDWARE INTERFACE

- ☐ Over internet, hardware should be connected to internet service
- ☐ Qualcomm Supportive

❑ PERFORMANCE REQUIREMENTS ~

○ EXPECTED SOFTWARE

- ☐ Cover all the devices running with Android versions greater than Android 6.0.
- ☐ Adapt to the requirements of latest kernels offered by newer versions of android (Currently Android 11).
- ☐ Friendly to pre-installed and post-installed launchers.

○ EXPECTED HARDWARE

- ☐ Able to work in devices with Processor frequency as low as 1.2GHz - 1.3GHz.
- ☐ Support Qualcomm and MediaTek every low end processor (like Snapdragon 450) to high end processors (Snapdragon 888) efficiently.
- ☐ Able to handle network traffic with the latest Wifi 6 system in flagship devices.

○ EXPECTED MEMORY USAGE

- ☐ Use not more than 1% of the RAM while they're in the background thread.
- ☐ In foreground, RAM usage must not exceed more than 1.5% of overall memory (eg. For a 4 GB RAM device memory consumption must not be more than 80 MB).
- ☐ Must not hang up, While Working With Other Applications

○ EXPECTED RUNTIME

- ☐ Delay not more than 2-3 seconds while logging in and out with around 10Mbps internet speed.
- ☐ Uploading time of the Query must be conditioned to resolution and number of images attached.
- ☐ A query with just texts must not take more than 10 sec to upload with the same upload speed as mentioned earlier.

❑ DESIGN CONSTRAINTS ~

○ USER PERSONAS

- ☐ Target Domain of Users is the Students Fraternity
- ☐ Apart From users , the admin will also be using it
 - Admins are for managing the database and for handling the objectionable queries.
- ☐ Involvement of developers , UI designers , architects is situation-sensitive.

○ ORIENTATION

- Design is Constrained for Vertical Orientation

○ USER FRIENDLINESS

- ☐ Effortless and fast authentication through Google Auth Service.
- ☐ Smooth operation to select images from device, formulate a write-up and post it in no time.
- ☐ Proffers an easy way to comment on the desirable query or post.

○ SOFTWARE ENVIRONMENT

- ☐ It's an android based software application
- ☐ Compatible with Android devices (*.apk*)
- ☐ Offers touch gesture based user interface
- ☐ Encompass interactive and dynamic medium to operate considering with certain mentioned functionalities

☐ NON-FUNCTIONAL REQUIREMENTS ~

With Reference to the ISO 25010 , Quality Attributes to be taken care of are :

○ LEARNABILITY

- ☐ Degree to which a product or system can be used by specified users to achieve specified goals of learning to use the product or system with effectiveness, efficiency, freedom from risk and satisfaction in a specified context of use.

○ USER INTERFACE AESTHETICS

- ☐ Degree to which a user interface enables pleasing and satisfying interaction for the user.

○ FUNCTIONAL SUITABILITY

- ▶ This characteristic represents the degree to which a product or system provides functions that meet stated and implied needs when used under specified conditions. This characteristic is composed of the following sub-characteristics:

☐ FUNCTIONAL COMPLETENESS

- ▶ Degree to which the set of functions covers all the specified tasks and user objectives.

☐ FUNCTIONAL CORRECTNESS

- ▶ Degree to which a product or system provides the correct results with the needed degree of precision.

☐ FUNCTIONAL APPROPRIATENESS

- ▶ Degree to which the functions facilitate the accomplishment of specified tasks and objectives.

○ MEASURES

COMPLETENESS :

- Wireframe the entire Project
- Wire framing allows to keep a check on what functionalities have been added and what not.

CORRECTNESS :

- It can be verified by allowing large number of user

- It'll check up if all intended users are being logged into their respective accounts
- By Checking the Output of the Action of a User and Tracing it back to see if the intended action is correct or not

APPROPRIATENESS :

- It can be measured by Checking out if the inter comment made up on a specific account has been notified to that person appropriately or is being send to somebody else
- It can also be measured with that, when some comment is added or [post is raised then it should be same as being typed by the user. It must be like the text had changed somehow

LEARNABILITY

- It can be check by testing with a number of volunteers & Getting their feedback on the experience with Software
- By User Experiences
- By User Journeys
- Getting all the pain points

USER INTERFACE AESTHETICS

- It can be check by testing with a number of volunteers & Getting their feedback on the experience with Software
- By User Experiences
- By User Journeys
- Getting all the pain points

☐ SCHEDULE & BUDGET ESTIMATES ~

○ TIMELINE ~ SCHEDULE

- ☐ **Project Duration \approx 2 Months**
- ☐ Project Initiation date ~ 15 March, 2021
- ☐ Testing phase ~ 3 May 2021 - 5 May 2021
- ☐ Release date ~ 7 May, 2021

○ BUDGET

☐ SALARY COMPUTATION :

- Front End developer for product design ~ 40LPA
- Backend Software Engineer ~ 20LPA
- Firebase Annual Subscription ~ Free

Hired Time = 2 Months (Approximate)

$$\text{Net Budget} = \left(\left(\frac{40}{12} \right) + \left(\frac{20}{12} \right) \right) * 2 \text{ LPA}$$

$$= 10,00,000 \text{ (10 Lakh) INR}$$