

# Negocio Conectar

CSD301 Project Report

INSTRUCTOR: Snigdha Biswas

Roll No	Names of Students
1610110145	Hari Sai Raghuram V
1610110236	Nitin Kumar
1610110261	Pranjal Sharma
1610110333	Samyak Bagra

---

# Content

Acknowledgement	3
About the project	4
Feasibility Study	6
Software Requirement Specification	9
Software Project Management	19
Design Documentation	26
Cost Estimation	33
Coding conventions	40
Software Testing Document	43
Future Enhancements	61

---

# Acknowledgement

The success and final outcome of this project required a lot of guidance and assistance from many people and we are extremely fortunate to have got this all along the completion of our project work. Whatever we have done is only due to such guidance and assistance and we would not forget to thank them. We owe our profound gratitude to our mentor Prof. Snigdha Biswas, who took keen interest on our project work and guided us all along, till the completion of our project work by providing all the necessary information for developing a good system. We are thankful to and fortunate enough to get constant encouragement, support and guidance from all Teaching staffs of Department of Computer Science which helped us in successfully completing our project work. Also, we would like to extend our sincere regards to all the non-teaching staff of department of computer science for their timely support.

---

# About the project

## Why this idea?

- A poor or inexistent business plan is the reason for many business failures, leading to their businesses being shut down as they don't attract the desired audience for their products.
- An effective business plan defines the goals of the business owner and formulates a road map to reach those goals.
- Also many startups fail due to lack of proper guidance and business plan.
- Businesses need a platform which helps them to withstand the market competition and gain profits.

## Business Statement

Our idea is to provide a platform for the small scale business providers and startups to run their business in a easy and comfortable environment, obtaining profits and running their businesses in the most efficient manner. We provide businesses with a platform to link-in to a community of support which stands by them through every aspect of their entrepreneurial journey.

- Advertise Jobs and Vacancies in respected communities
- Latest updates and news related to the particular industry
- Online Community to connect
- Obtain insight into the Statistics and Trends in the community

## Technical Requirement

- Backend: Java and firebase
- Frontend: Java (Android Studio)
- Platform: Application
- Model: Modified Waterfall Model.

---

## Why a Modified Waterfall Model?

- Simple and easy to understand and use
- Easy to manage as each phase has specific deliverables and a review process.
- Works well for smaller projects where requirements are very well understood.
- Clearly defined stages.
- Easy to arrange tasks.
- Implementation of easy areas does not need to wait for the hard ones.
- Process and results are well documented.
- Phases are allowed to overlap.

---

# Feasibility Study

## Technical Feasibility

1. Planned to use Android Studio and Firebase. All these are open source softwares and are available freely. No challenges in obtaining these softwares.
2. Since Negocio Conectar is a mobile application, testing can be done on different makes of phones available with the team members.
3. Since Negocio Conectar is a mobile application, testing can be done on different makes of phones available with the team members.

## Resource cost:

1. Entire team has required skills related to the tools and technologies used. Hence no external training required.
2. Entire team has high end laptops with good configuration. These can be used for development, testing and deployment purpose. Hence no challenges is involved in getting required hardware.
3. Time limitations for the product development and the ease of implementing these technologies are synchronized.

Negocio Conectar is Technically Feasible.

---

## Financial Feasibility

1. Since Negocio Conectar is a mobile application, it can be tested on various kinds of mobiles available with our team.
2. Since app does not use any multimedia data, it requires less operational bandwidth.
3. Planned to use Python, JavaScript, React Native, Node JS and MongoDB. All these are open source and do not incur any cost of licensing for development purpose following freeware software standards.

### Resource cost:

1. Entire team has required skills on tools and technologies used. Hence no dependency on external resources. Hence no associated cost for any resources.
2. Entire team has high end laptops with good configuration. These can be used for development, testing and deployment purpose. Hence there is no associated cost for hardware.
3. Being a mobile application Negocio Conectar will have an associate cost for hosting it online.

Negocio Conectar is Financially Feasible

## Legal Feasibility

1. Initially we named our product 'Business Connect', but we had to change the name of the application due to the copyrights for the name held by a different organisation.
2. We have renamed our application 'Negócio Conectar' and are in process of obtaining copyrights on this name.
3. We will have a non-disclosure agreement signed with the user whereby both parties cannot disclose information from the application.

---

## Operational Feasibility

1. We will have obtained API access from verified organisations ensuring reliability in the performance of the API.
2. We will have our members verified through Aadhar Verification.
3. We will incorporate a powerful verification algorithm which checks the content posted by the user and filters it accordingly.

## Scheduling Feasibility





---

# Software Requirement Specification

## Contents

### 1 Introduction

- 1.1 Purpose
- 1.2 Document Conventions
- 1.3 Intended Audience and Reading Suggestions
- 1.4 Project Scope
- 1.5 References

### 2 Overall Description

- 2.1 Product Perspective
- 2.2 Product Functions
- 2.3 User Classes and Characteristics
- 2.4 Operating Environment
- 2.5 Design and Implementation Constraints
- 2.6 User Documentation
- 2.7 Assumptions and Dependencies

---

### **3 External Interface Requirements**

3.1 User Interfaces

3.2 Hardware Interfaces

3.3 Software Interfaces

3.4 Communications Interfaces

### **4 System Features**

4.1 Community

4.2 Feeds

4.3 Advertisements

4.4 Search bar

4.5 Chat-bot

### **5 Other Nonfunctional Requirements**

5.1 Performance Requirements

5.2 Safety Requirements

5.3 Security Requirements

5.4 Software Quality Attributes

5.5 Business Rules

---

# 1 Introduction

## 1.1 Purpose

The purpose of this document is to present a detailed description of the mobile application Negócio Conectar. It will explain the purpose and features of the software, the interfaces of the software, what the software will do and the constraints under which it must operate. This document is intended for users of the software and also potential developers.

## 1.2 Document Conventions

This Document was created based on the IEEE template for System Requirement Specification Documents.

## 1.3 Intended Audience and Reading Suggestions

- Typical Users, such as Professionals, who want to use Negócio Conectar to connect with other professionals and hire potential employees.
- Unemployed individuals who are looking for potential jobs and connect with recruiting companies and organisations.
- Individuals looking for a better application which organises their tasks and make their professional lives comfortable and hassle free.

## 1.4 Project Scope

Negócio Conectar is a mobile application that people can use to connect with other people in the same profession. Users can use it find jobs and even offer job positions. This software helps users to find useful insights and trends in the industry and their profession's community. It has feeds with news from all over the internet related to the user's profession and industry. It also has a section for writing notes, a calendar marked with all the important schedules of a user, a reminder and AI based chatbot to answer all your queries related to your business problems.

---

## 1.5 References

<https://developer.android.com/studio/>

### **Contact:**

[hv829@snu.edu.in](mailto:hv829@snu.edu.in)

[ps549@snu.edu.in](mailto:ps549@snu.edu.in)

[sb472@snu.edu.in](mailto:sb472@snu.edu.in)

[ni333@snu.edu.in](mailto:ni333@snu.edu.in)

---

## 2 Overall Description

### 2.1 Product Perspective

Negócio Conectar provides a platform for the small scale business providers and startups to run their business in a easy and comfortable environment, obtaining profits and running their businesses in the most efficient manner. We provide businesses with a platform to link-in to a community of support which stands by them through every aspect of their entrepreneurial journey.

### 2.2 Product Functions

- Logging in and Signing up with Negócio Conectar
  - Login: Login to your account
  - Signup: Create your account with Negócio Conectar
  - Forgot Password: Reset your password after verifying that it is your account
- Home Screen for Negócio Conectar
  - Dashboard: The Dashboard which displays the latest features.
  - Searchbar: A Powerful Search bar which searches the whole community to provide related results.
  - ChatBot: A Smart AI based chatbot which provides key information and useful suggestions for users.
- Profile: Users can view and make changes to their profile.
- Advertisements Portal
  - View Advertisements: The users can view advertisements related to job offers and services.
  - Post Advertisements: The Users can post their own advertisements.
  - Fraud Search: An AI classifier which detects fraud job advertisements and reports it to the user.

- 
- News Feeds: An AI based algorithm classifies the news pulled from News providing APIs and only displays the appropriate and related News.

### **2.3 User Classes and Characteristics**

- Typical Users, such as Professionals, who want to use Negócio Conectar to connect with other professionals and hire potential employees.
- Unemployed individuals who are looking for potential jobs and connect with recruiting companies and organisations.
- Individuals looking for a perfect application which organises their tasks and make their professional lives comfortable and hassle free.

### **2.4 Operating Environment**

- Android Marshmallow and above

### **2.5 Design and Implementation Constraints**

Negócio Conectar is developed in JavaScript, it uses Flexbox and CSS for its Visualisation and has been developed on React-Native framework. It uses modular approach where every feature is wrapped into separate modules and the modules depend on one another. Many well defined APIs and Plugins are incorporate to make the working of the application simpler and faster.

### **2.6 User Documentation**

User documentation will soon be updated as soon as first phase of development is completed.

### **2.7 Assumptions and Dependencies**

Negócio Conectar has been developed for mobile operating systems. It can run on Android and iOS as these Operating Systems hold the highest market share for users. Negócio Conectar assumes that the mobile devices are using the latest release of the Operating System. As this application has been developed on JavaScript, Java must be installed on the devices for Negócio Conectar to work.

## **3 External Interface Requirements**

---

### 3.1 User Interfaces

Negócio Conectar Login Screen:

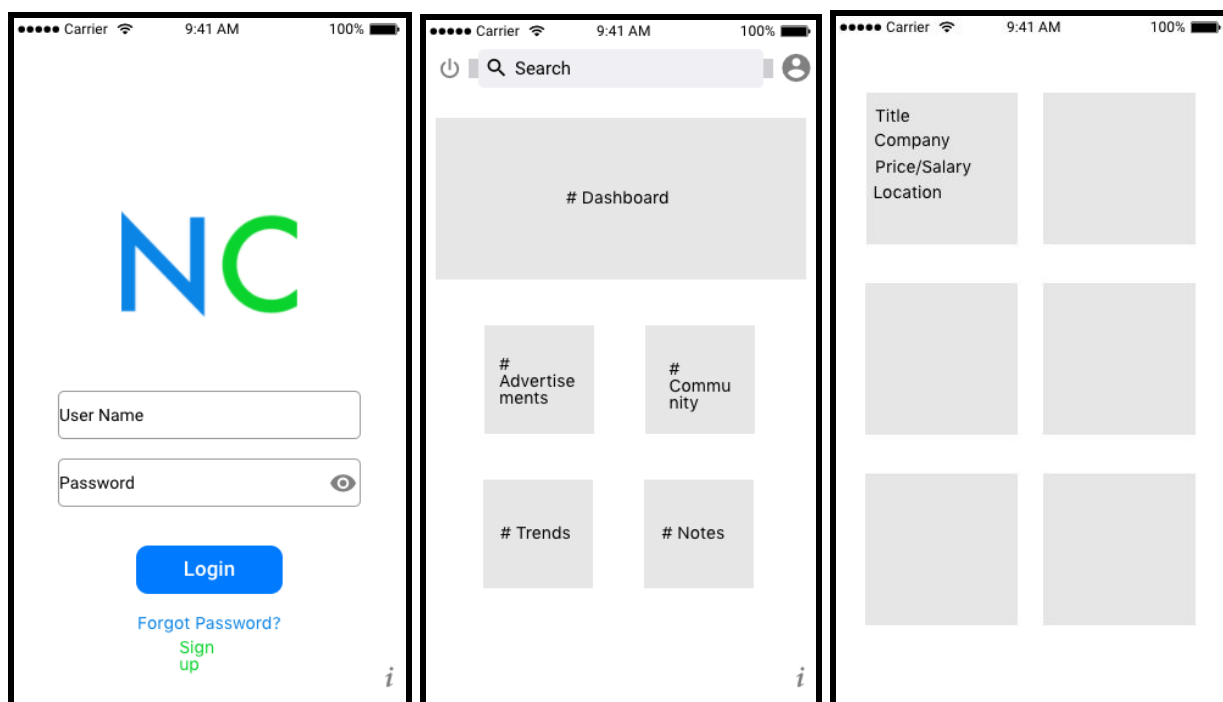
The Users willing to access Negócio Conectar must login into their registered accounts. The Password is secured and verified using AES.

Negócio Conectar Home Screen:

Users can access other features of Negócio Conectar through the main Home Screen. This screen provides the links to: Advertisements, Trends, Community, Notes and Reminders.

Negócio Conectar's Advertisements Portal:

Users can view and post opportunities through the advertisement portal.



---

### **3.2 Hardware Interfaces**

The minimum hardware requirements of Negócio Conectar are 128 MegaBytes of RAM and 256 MegaHertz CPU. Also as Negócio Conectar is an network based application it requires an active internet connection for it to be up and running. Basic features of the application can be used with these specifications. For faster and improved performance of the application improved network connectivity and hardware is required.

### **3.3 Software Interfaces**

Negócio Conectar requires Java to be installed on the system, more specifically Java version 7 or 8 for its latest release. Additional information can be found on section 2.7 of this document.

### **3.4 Communications Interfaces**

Negócio Conectar requires an active internet connection to connect with the application server to receive latest news and messages from the community. However the notes, calendar and planner can be used without connecting to the internet.

## **4 System Features**

This section demonstrates Negócio Conectar's most prominent features and explains how they can be used and the results provided to the user.

### **4.1 Community**

A portal for community where all the users belonging to a particular industry can post updates, suggestions and news related to the industry.

### **4.2 Feeds**

An Intelligent bot which provides the community with news which is related to their particular community.

### **4.3 Advertisements**



---

A portal for start-ups and companies to post jobs openings, vacancies or search for skilled professionals. The same portal is used by users to search for their desired positions and jobs.

#### **4.4 Search bar**

A powerful search bar to find content and information posted in various communities.

#### **4.5 Chat-bot**

An Intelligent interactive bot to help users with questions related to the app and related to their businesses.

## **5 Other Nonfunctional Requirements**

### **5.1 Performance Requirements**

There is no specific performance requirement except decent internet of user. So, good internet is required for faster page load times.

### **5.2 Safety Requirements**

To ensure that no user loses any private information on Negócio Conectar (due to a crash or any bug), we use redundancy techniques while storing data. There is report service through which users can report any crash or bug issues which will be fixed by our developers and release updates for the application.

### **5.3 Security Requirements**

---

To ensure that the users do not lose their account information, Negócio Conectar uses AES (Advanced Encryption Standards) to encrypt the login information from the users.

#### **5.4 Software Quality Attributes**

Negócio Conectar has been developed taking account the requirements of Business professionals in mind.

#### **5.5 Business Rules**

A typical user will be granted default privileges to use basic functional requirements while the administrative access is only allowed to developers. Also, data usage statistics generated on device will be sent to developers to improve the application and to fix bugs.

---

# Software Project Management

## 1. Introduction

### (a) Objective

Our idea is to provide a platform for the small scale business providers and startups to run their business in a easy and comfortable environment, obtaining profits and running their businesses in the most efficient manner. We provide businesses with a platform to link-in to a community of support which stands by them through every aspect of their entrepreneurial journey.

### (b) Major functions

Three major functions have been identified that the software will incorporate. They are:

- (i) Advertisements : A portal for startups and companies to post jobs openings, vacancies or search for skilled professionals. The same portal is used by users to search for their desired positions and jobs.
- (ii) News : An AI based news classifier which gets the latest news.
- (iii) ChatBot : An interactive AI based chatbot which gives recommendations to the users about their queries and business problems.

.

### (c) Performance Issues

The following performance issues could occur:

- 1) A good advertisement of headline not being listed because it was not in the top results returned by a the app.
- 2) Since the software relies on the results returned by news agencies which is a third party, any slowdown or site crash will stall the the section as well.
- 3) In a traffic spike, new users will not be able to sign in.

---

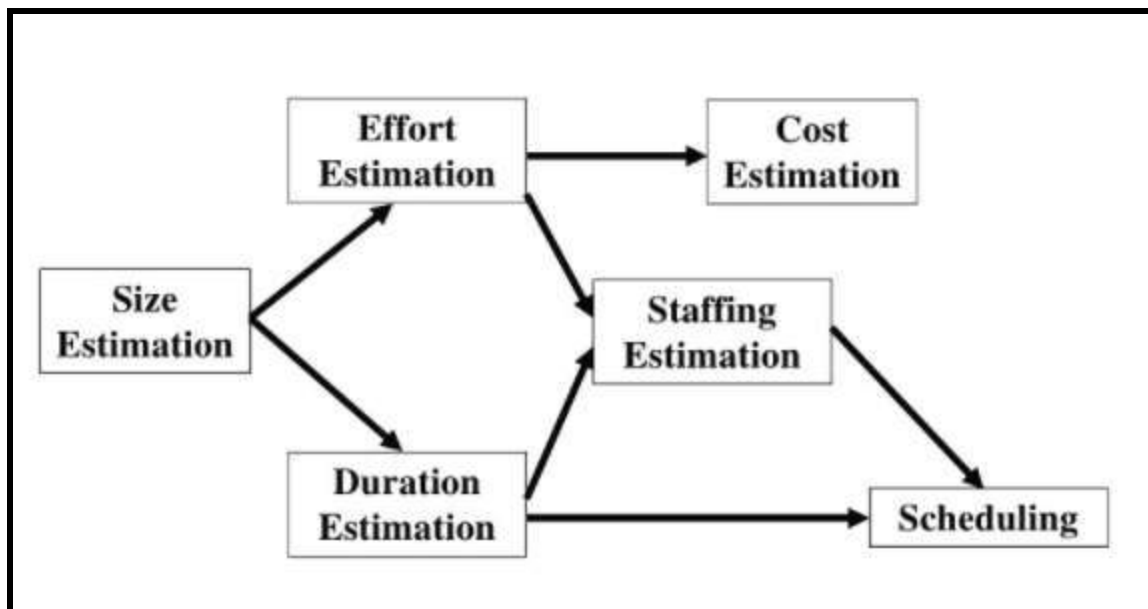
## 2. Project Estimates

### (a) Historical Data Used

We have tried to base the project requirements and functions in accordance with our prior knowledge. The individual capabilities are also kept in mind while defining team roles. Another kind of historical data used is the user base of similar existing platforms. The popularity and extensive use of such platforms has affected the requirements and functionalities of this project.

### (b) Estimation Technique Used:

- We formed our estimations as per the following flowchart:



- The basis of our estimation i.e. Size Estimation will be done using the Function Point (FP) technique.

### (c) Effort, Resource, Cost and Project duration techniques:

- Efforts - A dedicated team of 4 people will work together to handle designing, coding, testing and documents.

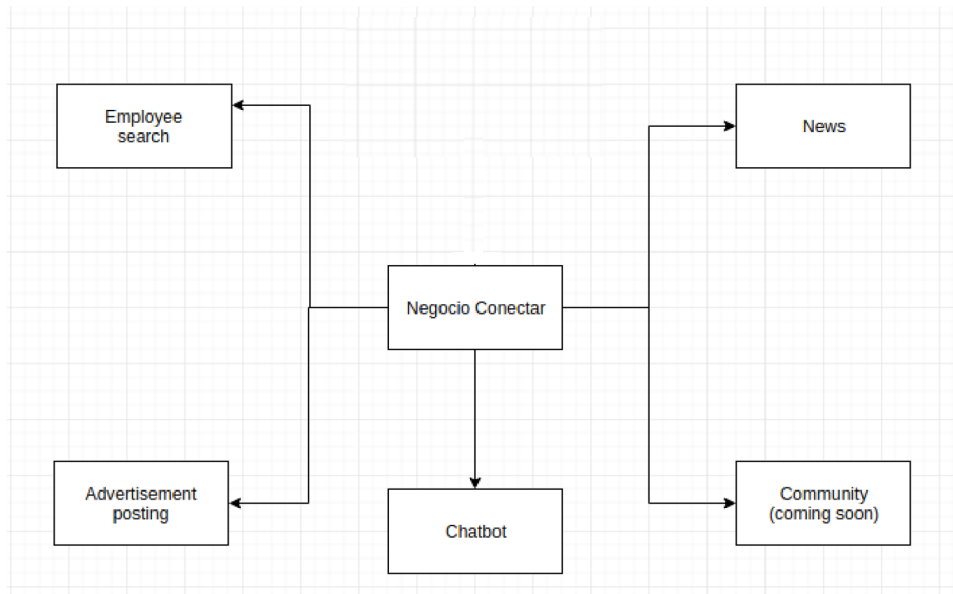
- 
- Resources - The technical resources comprises of firebase software along with java necessary for deployment.
  - Cost - Since at present the project is not available for a large number of users, cost is negligible. Also there is no requirement of any expensive hardware.
  - Duration - This project is expected to finish within 3-4 months.

### **3. Scheduling**

Scheduling is an important activity for all the project managers.

#### **(a) Task Structure**

- The tasks needed to complete the project:
  - Requirement Analysis
  - Design
  - Build Prototype
  - Testing
  - Documentation
- The dependency among different tasks.
  - Building Prototype depends upon the Requirement Analysis and Design of the product and the documentation
  - Testing phase depends majorly upon the prototype built and rest of the phases also
- The estimates for the duration of the identified tasks are:
  - Feasibility Study: August
  - Project Planning: August end
  - Requirement Analysis: mid September
  - Design: September end
  - Building Prototype: October to November
  - Testing: November mid
  - Delivery of product: November end



## 4. Project Resources

### (a) Roles and Responsibilities-

The main responsibilities involve coding & application design, database management, testing and documentation of the process.

The roles of each member of the team are defined as follows-

(i) Team Leader- Samyak Bagra

- Motivate the team members to perform their tasks
- Help the team in allocating the tasks and resolving issues
- Creates and maintains the SRS
- Java Dynamic web applications for application platform development

(ii) Development Manager- Hari Sai Raghuram V

- Lead the team in producing the development strategy
- Lead the team in producing the preliminary size and time estimates for the products to be produced
- Lead the team in producing the design specification
- Lead the team in coding and design

---

(iii) Process Manager- Nitin Kumar

- Establish and maintain the team's development standards
- Act as the team's inspection moderator
- Handles back-end database management
- Lead the team in producing the balanced team plan

(iv) Design Manager- Pranjal Sharma

- Lead the team in producing and tracking the quality plan
- Alerts the team to quality problems
- Manage the configuration management system
- System testing plan

Documentation will be handled by all members.

## **(b) Software and Hardware**

(i) Software

- Back-end- Java(Firebase)
- Front-end- Java(Android Studio)
- Platform- Application
- Model- Modified Waterfall model

(ii) Hardware:

To accommodate the database, storage space is used.

## **5. Staff Organization**

### **(a) Team Structure**

The development team consists of 4 members each with the following roles:

Team Leader and Front end (Java) - Samyak

---

Back end and DBMS - Raghuram

Coding and application development- Nitin

Design and testing- Pranjal

Documentation- Will be handled by each member in rotation

## **(b) Management Reporting**

Every meeting that the team has is documented and circulated amongst the members as the Minutes of the Meeting. After the planning stage, the SRS document will be drawn up as well to act as a contract between the developers and the client. Once the coding starts, documentation will be maintained and there will be a user's manual and a troubleshooting manual that the developers will include with the software. All this documentation of important information in a systematic manner will ensure that the project is on schedule and that any problems faced during the software development process are dealt with quickly and efficiently.

## **6. Risk Management Plan**

### **(a) Risk Analysis:**

Risk analysis is done throughout the meetings.

### **(b) Risk Identification:**

- Project Risks – The only viable risk associated with our project can be the personnel limits. Our knowledge might impose risks later and restrict the utility of the project.
- Technical Risks – We use a lot of third party api's to get relevant information to our users. Their services also affect our services.
- Business Risks – There are some applications which run on similar concept, these applications will give us competition

### **(c) Risk Mitigation Procedure:**



---

We are using a modified waterfall model, we will try to mitigate the arriving risks on our working model after every phase. This procedure will ensure a valid working product at the end.

---

# Design Documentation

## 1. Introduction

### i. Purpose and scope

Purpose of this document is to give a complete yet concise description of the design of the “Negocio Conectar” app , including software, hardware and User Interface. It also gives clear guidelines to the client and developers about what functionality will be incorporated into the system.

Components covered in this document include:-

- Product overview
- Detailed functional requirements
- Interface
- Miscellaneous details

### ii. Product and Environment

Product Purpose and Goal:

Our idea is to provide a platform for the small scale business providers and startups to run their business in a easy and comfortable environment, obtaining profits and running their businesses in the most efficient manner. We provide businesses with a platform to link-in to a community of support which stands by them through every aspect of their entrepreneurial journey.

Environment:

- Application platform with GUI from Android Studio.
- Java, firebase for database in backend.

---

### **iii. Definitions, Acronyms and Abbreviations**

GUI – Graphical User Interface

MB – Mega Byte

GB – Giga Byte

RAM – Random Access Memory

API – Application Program Interface

OOP – Object Oriented Programming

### **iv. References**

1. <https://www.w3schools.com/>
2. <https://docs.oracle.com/javase/7/docs/api/>

## **2. Overview of the System**

### **(i) Application Environment**

Our application consists of different components and android studio is consists of resources required for developing and hosting the application. Resources include java for development and firebase for development of web components like news, ads and storing other details.

### **(ii) Interface**

The GUI will be pretty basic, enabling the users to easily traverse through the application and understanding functionality of each component.

### **(iii) Hardware (environment)**

---

Cellular device having internet connection.

**(iv) Software (environment)**

Android Marshmallow or above is required to run the application.

### **3. System Architecture**

#### **Architectural Design**

(a) General Guidelines

There are number of subdivisions present in the application.

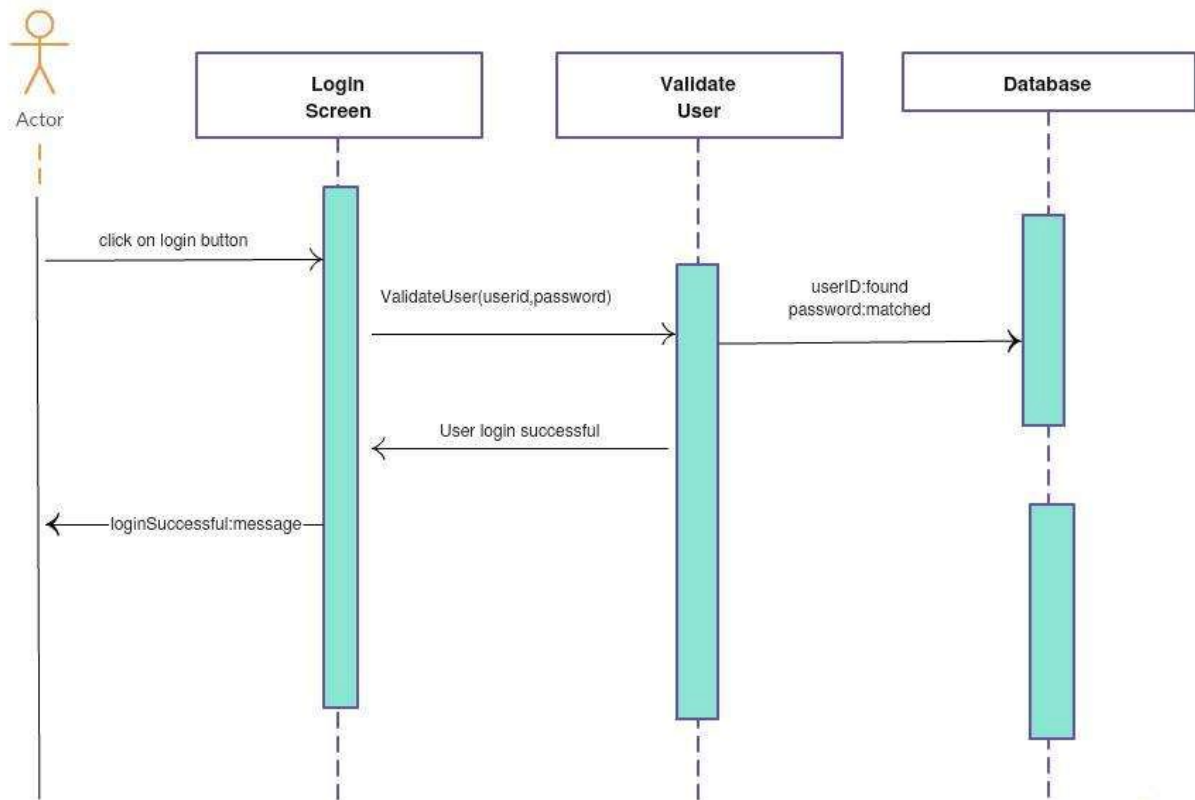
The following project modules are:

- Login
- Sign Up
- Advertisements
- News
- Post Advertisement
- Chat-bot

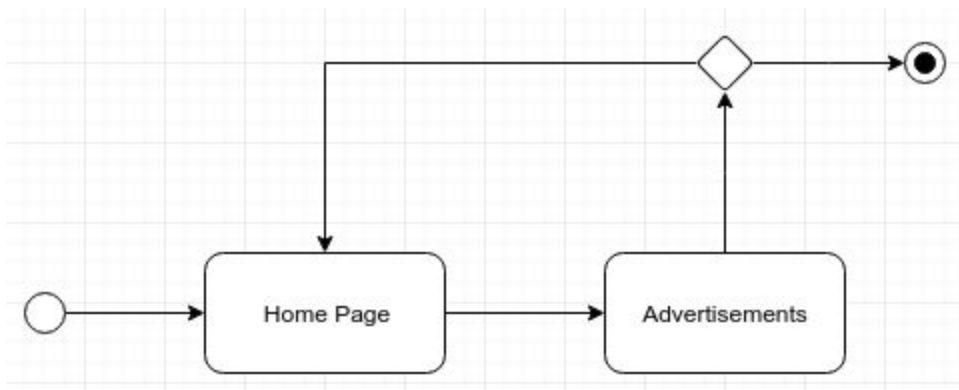
(b) Decomposition Description

This is a decomposition of the subsystems in the architectural design.

i) Sequence Diagram : User Login



ii) Activity Diagram : View Advertisements



---

### (c) Architecture view

We decided to use Shared Data Style as our c&c view for our project. It offers several advantages -

- System stores the shared data in- Data repositories – could be File system or database
- Provides a reliable and permanent storage
- Takes care of the concurrent data access
- Provides data access support

System stores the shared data in- Data repositories – could be File system or database

Data accessors -

- Access the data from the repositories
- Perform computation on the data obtained
- If they want to share the result with other components, put the result back in the repository
- These components do not directly communicate with each other, data repository components are the means of communication between them.

More specifically we will use repository style, as it is passive repository of data, provides permanent storage and related control for data accessing.

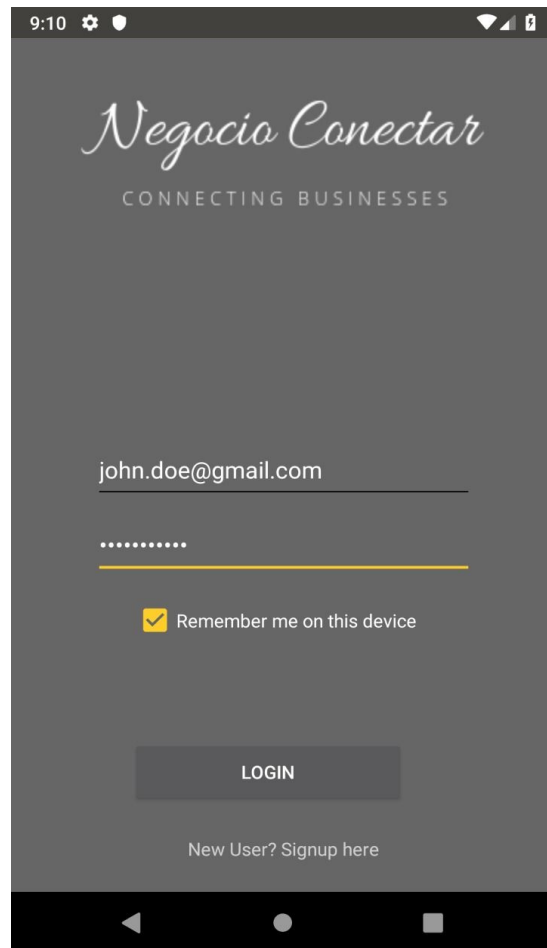
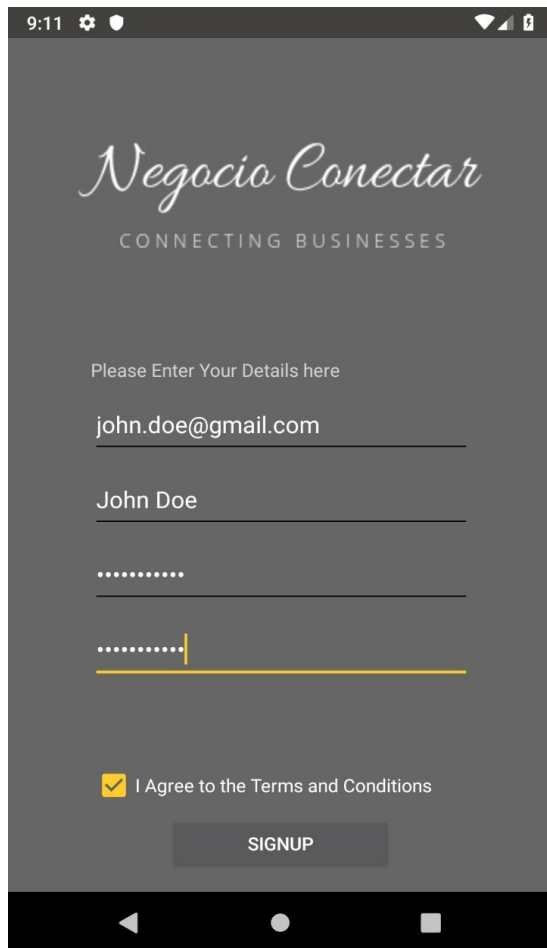
## 4. Data Design

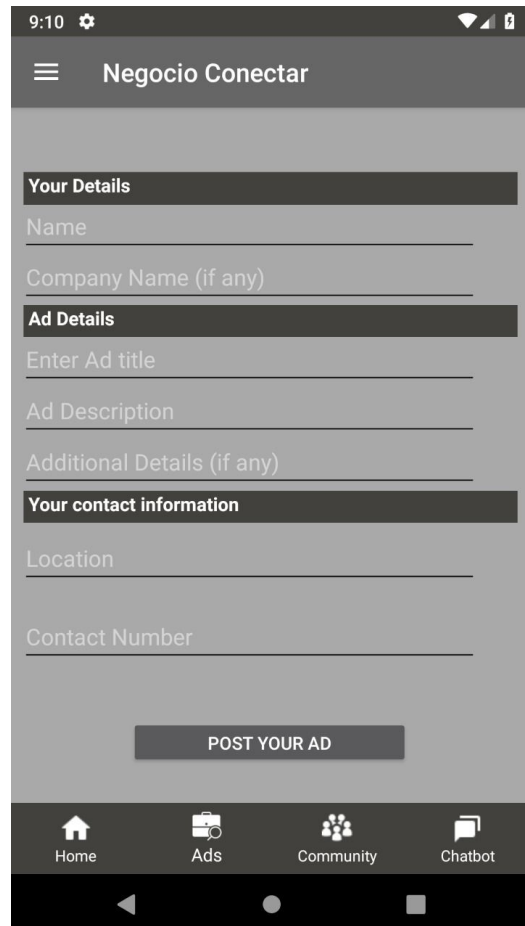
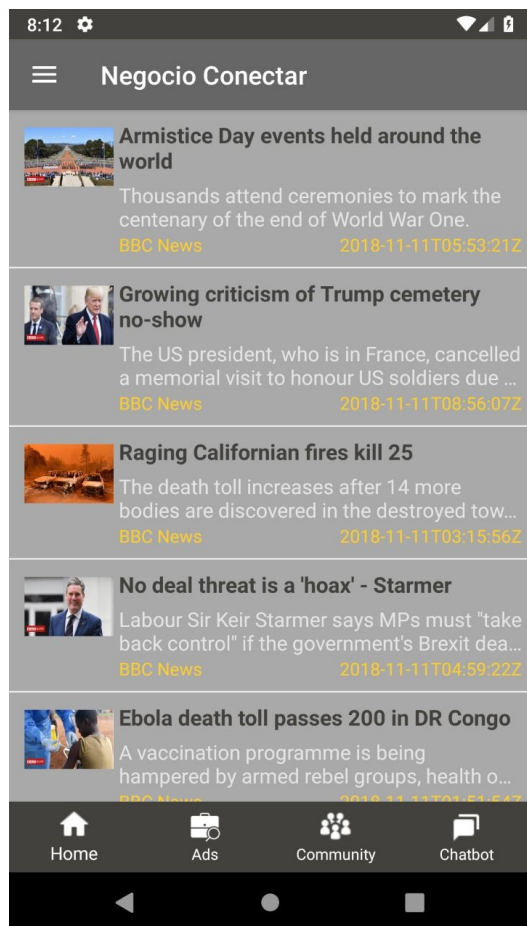
### 4.1) Data Description

- The login details are the 1st input. They are stored in our database, and are cross checked from database to authorise login.
- Then there is option for sign up, in which case once user sign up, they get a confirmation email on their registered mail id.

- The data from the application is fetched and stored in our database, for which we incorporated Firebase using Java.

## 5. Screen Images







---

# Cost Estimation

In order to achieve efficient and effective management of software projects, it is important to estimate the size and cost of the project. For the size and cost estimation of our SE project, we have used the Function Point Analysis (FPA) and COCOMO model to predict the development effort of the statistical analysis package. Function point analysis and the COCOMO model are an accepted standard for the measurement of software size in software engineering.

The FPA is a reliable method for measuring the size of computer software. It essentially measures functionality that the user requests and receives. It also measures the software development and maintenance cost and size independently of the technology used for implementation.

## **The general approach that FPA follows is**

1. Count the number of inputs, outputs, inquiries, master files, and interfaces required, then calculate the Unadjusted Function Points (UFP)
2. Calculate the adjusted function point (AFP) by multiplying these counts by an adjustment factor; the UFP and the product complexity adjustment.
3. Calculate the Source Lines of Code (SLOC) with the help of the AFP and the Language Factor (LF)

---

### Calculation of Unadjusted Function Points ( UFP )

Function Type	Simple	Average	Complex
External Input	3	4	6
External Output	4	5	7
External Inquiry	3	4	6
Internal Logical File	7	10	15
External Interface File	5	7	10

		Weighting Factor			Count
		Simple	Average	Complex	
Inputs	Member Login		4		32
	Member Registration	3			
	Search bar			6	
	Notes and reminders	3			
	Posts (in community)	3			
	AI Chat			6	
	Advertisement (insert)		4		
	Advertisements (search)	3			

---

		Weighting Factor			Count
		Simple	Average	Complex	
Outputs	News	4			31
	Community Posts	4			
	Notes and Reminders	4			
	Advertisements		5		
	Trends			7	
	AI Chat			7	
Inquiries	Validate Member Information	3			3

		Weighting Factor			Count
		Simple	Average	Complex	
Files	User Records and Statistics		10		31
	Advertisements	7			
	Community	7			
	News	7			
Interfaces	User to Application Server		7		17
	News API	5			
	Firebase	5			
Total UFP		58	30	26	114

---

### Calculate Adjusted Function Point

Number	Complexity Weighting Factor	Value
1	Backup and Recovery	4
2	Data Communications	3
3	Distributed Processing	2
4	Performance critical	5
5	Existing operating environment	4
6	Online data entry	3
7	Input transactions over multiple screen	0

Number	Complexity Weighting Factor	Value
8	Master files updated online	0
9	Information domain values complex	4
10	Internal Processing complex	2
11	Code designed for reuse	2
12	Conversion/installation in design	0
13	Multiple installations	4
14	Application designed for change	2
	<b><u>Total complexity adjustment value</u></b>	35

---

### 3. Calculate the Source Lines of Code (SLOC) and the formulas used

- Total Unadjusted Function Points (UFP) = 114
- Product Complexity Adjustment (PC) =  $0.65 + (0.01 * 35)$  = 1.00
- Total Adjusted Function Points (FP) =  $UFP * PC$  = 114
- Language Factor (LF) for Java assumed as = 38
- Source Lines of Code (SLOC) =  $FP * LF$  = 4332

## COCOMO Model

- The Constructive Cost Model (COCOMO model) is a procedural cost estimation model.
- The model parameters are derived from fitting a regression formula using data from historical projects.
- Basic COCOMO computes software development effort and effort as a function of program size.
- Program size is expressed in estimated thousands of source lines of code (KLOC)  
COCOMO applies to 3 classes of software projects -
  - Organic Projects - Small teams with Good experience working in a 'not very rigid' environment.
  - Semi-Detached Projects - "Medium" teams with mixed experience working with a mix of rigid and not so rigid requirements.
  - Embedded Projects - Developed with a set of "tight" constraints. It is also a combination of Organic and Semi-Detached projects.

---

VL	-	Very low
L	-	Low
N	-	Normal
H	-	High
VH	-	Very High
XH	-	Extra High

**Product Attributes**

Required Reliability	1.40 (VH)
Database Size	1.08 (H )
Product Complexity	1.00 (N )

**Computer Attributes**

Execution Time Constraint	1.00 (N )
Main Storage Constraint	1.00 (L )
Platform Volatility	0.87 (L )
Computer Turnaround Time	1.00 (N )

**Personnel Attributes**

Analyst Capability	0.86 (H )
Applications Experience	0.91 (H )
Programmer Capability	0.86 (H )
Platform Experience	1.00 (N )
Programming Language and Tool Experience	1.00 (N )

**Project Attributes**

Modern Programming Practices	1.00 (N )
Use of Software Tools	0.83 (VH)
Required Development Schedule	1.10 (VH)

**New (Values are probably wrong)**

Required reusability	1.00 (L )
Documentation match to life-cycle needs	1.00 (N )
Personnel continuity	1.00 (N )
Multisite development	1.00 (L )

COCOMO RESULTS for Language Factor								
MODE	"A" variable	"B" variable	"C" variable	"D" variable	KLOC	EFFORT, (in person-months)	DURATION, (in months)	STAFFING, (recommended)
semi-detached	2.4249420853257604	1.12	2.5	0.35	4.332	12.525	6.056	2.068
<p>Explanation: The coefficients are set according to the project mode selected on the previous page, (as per Boehm). The final estimates are determined in the following manner:</p> <p><b>effort</b> = <math>a \cdot KLOC^b</math>, in person-months, with KLOC = lines of code, (in thousands), and:</p> <p><b>staffing</b> = effort/duration</p> <p>where a has been adjusted by the factors:</p>								

Website : <https://strs.grc.nasa.gov/repository/forms/cocomo-calculation/>

---

# Coding conventions

Coding conventions are a set of guidelines for a specific programming language that recommend programming style, practices, and methods for each aspect of a program written in language.

Coding conventions are necessarily required by the pieces of code for better understanding, coordination and maintainability of the software application.

## Languages

We have used Java for backend and front-end.

## Naming conventions

Java files with name as per their function for better readability of code.

## Beginning Comments

Files included with comments explaining the contents of file, whenever required.

## Comment conventions -

1.Algorithmic description - Algorithms explained whenever required, in the form of comments.

A screenshot of a code editor showing Java code. The code defines a private method `searchNameStringRequest` that takes a `String nameSearch` parameter. The method contains several final String variables for API endpoints, search parameters, and sorting options. A comment at the bottom explains the parameters of the method: `// 1st param => type of method (GET/PUT/POST/PATCH/etc)`, `// 2nd param => complete url of the API`, and `// 3rd param => Response.Listener -> Success procedure`.

```
private RequestQueue newsQueue;

private StringRequest searchNameStringRequest(String nameSearch) {
    final String API = "fe0e554c0ed24a8da06134a60cffa2f4";
    final String NAME_SEARCH = "&q=";
    final String DATA_SOURCE = "&ds=Standard Reference";
    final String FOOD_GROUP = "&fg=";
    final String SORT = "&sort=r";
    final String MAX_ROWS = "&max=25";
    final String BEGINNING_ROW = "&offset=0";
    final String URL_PREFIX = "https://newsapi.org/v2/top-headlines?country=india&";

    //String url = URL_PREFIX + API + NAME_SEARCH + nameSearch + DATA_SOURCE + FOOD_GROUP + SORT + MAX_ROWS + BEGINNING_ROW;
    String url = URL_PREFIX + API;

    // 1st param => type of method (GET/PUT/POST/PATCH/etc)
    // 2nd param => complete url of the API
    // 3rd param => Response.Listener -> Success procedure
```

2.Code description - Code consists of necessary explanation required as per the feasibility for the coders.



---

```

//flipper for main screen
public void flipperImages(int image){
    ImageView imageView = new ImageView(getActivity());
    imageView.setBackgroundResource(image); //backgrounds for flipper
    v_flipper.addView(imageView);
    v_flipper.setFlipInterval(4000);    //flip interval = 4 sec
    v_flipper.setAutoStart(true);       //start as soon as activity starts
    //setting up animation
    v_flipper.setInAnimation(getActivity(), android.R.anim.slide_in_left);
    v_flipper.setOutAnimation(getActivity(), android.R.anim.slide_out_right);
}

```

3. Debugging comments - comment used for debugging or to check different cases.

```

    if (model.getMsgUser().equals("user")) {
        holder.rightText.setText(model.getMsgText());

        holder.rightText.setVisibility(View.VISIBLE);
        holder.leftText.setVisibility(View.GONE);
    }
    else {
        holder.leftText.setText(model.getMsgText());

        holder.rightText.setVisibility(View.GONE);
        holder.leftText.setVisibility(View.VISIBLE);
    }
}

@NonNull
@Override
public chat_rec onCreateViewHolder(@NonNull ViewGroup viewGroup, int i) { return null; }
}

adapter = new FirebaseRecyclerAdapter<ChatMessage, chat_rec>(ChatMessage.class, R.layout.msglist, chat_rec.class, ref.child("messages")) {
    @Override
    protected void populateViewHolder(chat_rec viewHolder, ChatMessage model, int position) {

        if (model.getMsgUser().equals("user")) {

            viewHolder.rightText.setText(model.getMsgText());

            viewHolder.rightText.setVisibility(View.VISIBLE);

```

## Exception handling

Our project handles the following errors-

---

Error if register with

- invalid email
- invalid password while logging in
- using unknown firebase username and password for log in, that is not already registered.

### **Indentation style**

We have used *Ratliff style* for indentation of brackets. It consists of opening brace in same line as block name and closing brace after completion of the block. Nested blocks also follows same style.

---

# Software Testing Document

## Purpose

The purpose of the test plan is to ensure the proper working of the system, such that it is thoroughly tested, resulting in a successful implementation of new functionality, interfaces. The test plan will make sure that it covers all the functionalities with the available resources.

## Overview

### Product Name:

Negocio Conectar

### Project Leads-

- **Project Manager:**  
Samyak Bagra
- **Development Lead:**  
Hari Sai Raghuram
- **Test Lead:**  
Nitin Kumar
- **Test Case Designers:**  
Pranjal Sharma

### Test Personnel:

Test personals include the entire development team. Each module will be tested by both developer as well as another team member who has not worked on the module will test module.

### Documentation reviewers:

The Documentation reviewers include the entire development team. A team member who has not drafted it initially will review each document at least twice.

---

## **Product Review**

### **Project References**

The following are the list of the references that were used in preparation of this document.

- SRS Document

## **Testing Synopsis**

### **Items to be tested**

The system will be tested for all the functional requirements specified in the SRS documents. The user interface for the system will also be tested.

### **Items not to be tested**

Any requirements not specified in the SRS documents will not be tested.

### **System Requirements**

Client Requirements

System Requirements

### **Standard/Reference materials**

Software Requirement Specification (SRS)

---

## TYPES OF TESTING

### Test Type Definition

Test Type	Definition
<b>Unit Testing</b>	First level of testing is unit testing-for verification of code produced by the programmer
<b>Integration Testing</b>	Many units of tested modules are combined, which are then tested.
<b>Requirements Testing</b>	Verifies the specific requirements of the users are met.
<b>Acceptance Testing</b>	<p>The next level is system testing and acceptance testing – Entire software system is tested.</p> <ul style="list-style-type: none"><li>● Reference documents for this process is Requirement document.</li><li>● Acceptance testing is performed with realistic data</li></ul>
<b>Regression Testing</b>	Another level of testing call. Regression testing is performed when some changes are made to an existing system.
<b>User Interface(GUI) Testing</b>	Verify UI features and elements

---

**FEATURE LEVEL TESTING**

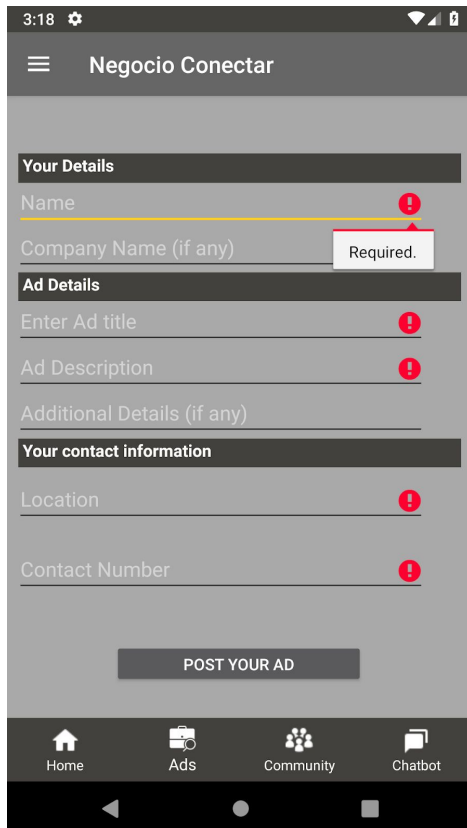
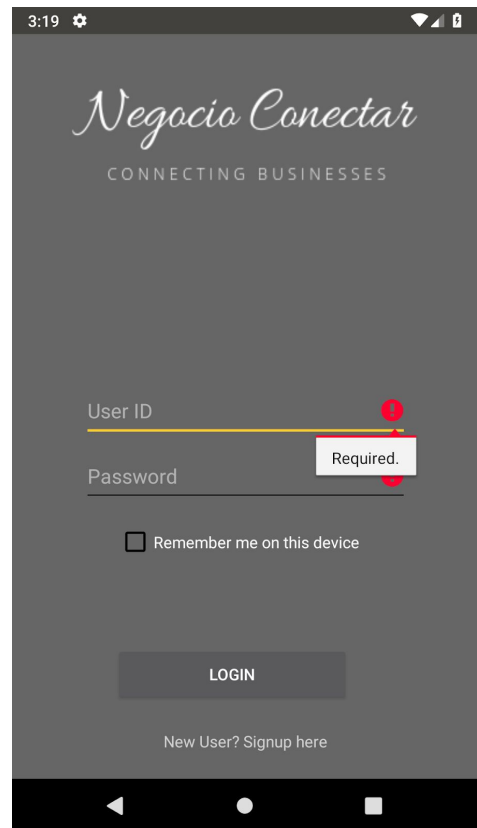
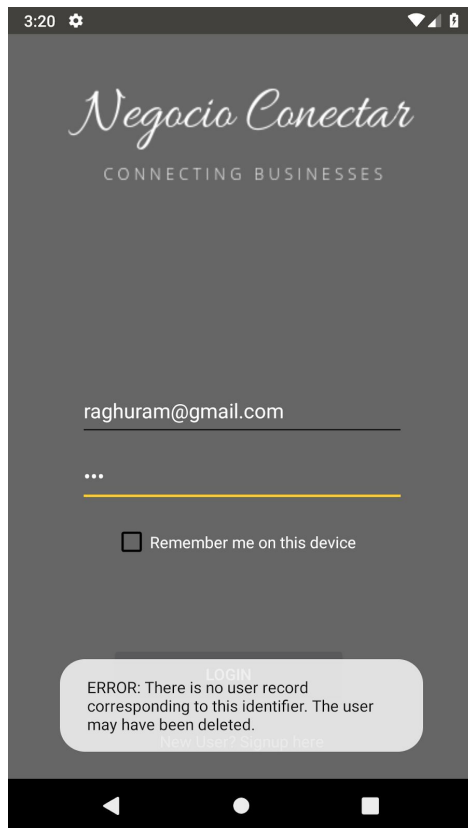
Functional Requirement Testing	Unit Testing	System Testing	Regression Testing	Acceptance Testing	GUI Testing
Login and Registration	Y	Y	Y	Y	Y
News	Y	Y	Y	Y	Y
Advertisement	Y	Y	Y	Y	Y
Home	Y	Y	Y	Y	Y
Profile and Information	Y	Y	Y	Y	Y

---

### Forced-Error Tests

System will be tested by entering incorrect data to check that the system show appropriate errors. Some of the errors can be as follows:

Error Condition	Messages
Invalid Username or Password	Invalid username or password
If the text field of username or password is left empty	Username/Password Required
Incomplete advertisement posting submitted	The form is not complete





---

### **Real World User-Level Testing**

Once the System level test is completed entering dummy data will carry out Real World User-Level Test.

### **Regression testing**

After each module is completed regression testing will be carried out. Also after Integration testing the Regression testing will be carried out for the entire system.

### **Configuration and compatibility testing**

Testing of the system has been done for the following  
Android 6.0 (Marshmallow)

### **Documentation testing/online help testing**

Following documents for testing will be submitted

- Test Plans
- Test Cases

### **Copyrights and license agreement**

The software is developed for the institute.

---

## Test Schedule and resources

Testing phase will start from:

Phases	Planned Start Date	Actual Start Date	Planned End Date	Actual End Date
Implementation	12/08/18	15/08/18	14/11/18	12/11/18
Unit Testing	12/11/18	10/11/18	25/11/18	15/11/18

## Unresolved issues and risks

If testing is taking longer than the time allotted in the project plan, the core project team will determine the possible options (more time) or team may even limit the scope of testing. The project guides will be notified. We ensure that every feature is tested as per document planning.

A test case is a document , which has a set of data inputs, expected results developed for a particular test scenario in order to verify compliance against a specific requirement.

<b>TEST CASE TITLE</b>	<b>REGISTER A USER</b>
<b>CUSTOMER REQUIREMENT/PURPOSE</b>	Only users with valid e-mail ids should be registered.
<b>PREREQUISITES</b>	User must have their respective e-mails and provide a password..
<b>STEPS</b>	<ul style="list-style-type: none"> <li>· Go to Registration Page</li> <li>· Enter a valid e-name</li> <li>· Enter the name</li> <li>· Enter a password</li> <li>· Repeat the password</li> <li>· Agree to the terms and Conditions</li> <li>· Click on Sign Up button</li> <li>· Check the profile page</li> </ul>
<b>EXPECTED RESULT</b>	User gets directed to the home page of the app and has a user profile page displaying his basic info and asking him/her the option he wants to choose next.
<b>TEST INPUT DATA</b>	Name, Mail id, Password
<b>OUTPUT</b>	The app opens showing the users profile details in the profile page
<b>STATUS</b>	PASS

---

<b>TEST CASE TITLE</b>	<b>LOG INTO THE APPLICATION</b>
<b>CUSTOMER REQUIREMENTS/PURPOSE</b>	A user can login if and only if he has already registered and his entry is stored in the database
<b>PREREQUISITES</b>	<ul style="list-style-type: none"> <li>● The user should be registered with the application</li> <li>● The database should be working in the backend to verify the user's credentials</li> <li>● There should be a connection between the front end and the back end</li> </ul>
<b>STEPS</b>	<ul style="list-style-type: none"> <li>● Enter your username and password</li> <li>● Click on OK</li> </ul>
<b>EXPECTED RESULT</b>	The user should be redirected to the home page of the application
<b>TEST INPUT DATA</b>	Mail id, Password
<b>OUTPUT</b>	The app opens showing the users profile details in the profile page
<b>STATUS</b>	PASS
<b>NOTES</b>	If the user enters a wrong email or wrong password (both verified from the database), an error message is generated and then the user is asked to enter them again.

---

<b>TEST CASE TITLE</b>	<b>EMPTY FIELDS DURING LOGIN/ SIGN UP</b>
<b>CUSTOMER REQUIREMENTS/PURPOSE</b>	None
<b>PREREQUISITES</b>	<ul style="list-style-type: none"> <li>● The database should be working in the backend to verify the user's credentials</li> <li>● There should be a connection between the front end and the back end</li> </ul>
<b>STEPS</b>	<ul style="list-style-type: none"> <li>● Do not enter any username and password during login or do not enter any user details during sign up.</li> <li>● Click on login/sign up button</li> </ul>
<b>EXPECTED RESULT</b>	An error message stating "Please enter the details" should be displayed.
<b>TEST INPUT DATA</b>	None
<b>OUTPUT</b>	An error red icon is displayed.
<b>STATUS</b>	PASS
<b>NOTES</b>	In order to make the application secure , the user needs to enter his valid credentials, otherwise the application won't allow the user to enter the application and shop with just pressing the SIGN UP or the LOG IN button.

---

<b>TEST CASE TITLE</b>	<b>VALIDATION OF E-MAIL</b>
<b>CUSTOMER REQUIREMENTS/PURPOSE</b>	In order to use the application, the user needs to enter his valid e-mail address.
<b>PREREQUISITES</b>	<ul style="list-style-type: none"> <li>● A working internet connection.</li> <li>● Connection between the backend(database) and the front end must be active.</li> </ul>
<b>STEPS</b>	<ul style="list-style-type: none"> <li>● Enter your e-mail address</li> </ul>
<b>EXPECTED RESULT</b>	If the e-mail address is not registered with any of the e-mail companies and is a fake one, then the user should not be able to sign up.
<b>TEST INPUT DATA</b>	E-mail address of the user
<b>OUTPUT</b>	The user is able to register irrespective of the correctness of the e-mail address.
<b>STATUS</b>	FAIL
<b>NOTES</b>	The security of the application is at stake if the user enters an invalid e-mail address and if he is successfully able to get registered and use the app.

---

<b>TEST CASE TITLE</b>	<b>VALIDATE A USER</b>
<b>CUSTOMER REQUIREMENTS/PURPOSE</b>	A user can login on the application only if he/she is registered
<b>PREREQUISITES</b>	● User should have a working internet connection.
<b>STEPS</b>	● Go to the login page of the app ● Enter correct mail id and password ● Click on the login button
<b>EXPECTED RESULT</b>	All the user details-name, username, password, e-mail id should be displayed in the database
<b>TEST INPUT DATA</b>	E-mail id and password of the user
<b>OUTPUT</b>	An error message saying that either the username or password is incorrect
<b>STATUS</b>	PASS
<b>NOTES</b>	This is a case of forced error testing.

---

<b>TEST CASE TITLE</b>	<b>VALIDATION OF DATABASE</b>
<b>CUSTOMER REQUIREMENTS/PURPOSE</b>	None
<b>PREREQUISITES</b>	<ul style="list-style-type: none"> <li>● Internet Connection</li> <li>● Check Firebase connections</li> </ul>
<b>STEPS</b>	<ul style="list-style-type: none"> <li>● Open the firebase website</li> <li>● Check the table in firebase website</li> </ul>
<b>EXPECTED RESULT</b>	When user enters incorrect mail id or password, error must be displayed.
<b>TEST INPUT DATA</b>	Tally from Firebase Table.
<b>OUTPUT</b>	All the user details are displayed in the database
<b>STATUS</b>	PASS



---

<b>TEST CASE TITLE</b>	<b>ADVERTISEMENT PAGE</b>
<b>CUSTOMER REQUIREMENTS/PURPOSE</b>	The user is able to see all the advertisements posted.
<b>PREREQUISITES</b>	<ul style="list-style-type: none"><li>● A working internet connection</li><li>● User must be logged in</li><li>● User opens the advertisement page by selecting the button on the home page.</li></ul>
<b>STEPS</b>	<ul style="list-style-type: none"><li>● Click on the advertisement button on the home page.</li></ul>
<b>EXPECTED RESULT</b>	The user should be able to see all the advertisements posted on the app.
<b>TEST INPUT DATA</b>	None
<b>OUTPUT</b>	A page showing all the advertisements posted on the app opens up.
<b>STATUS</b>	PASS

TEST CASE TITLE	ADVERTISEMENT FORM
CUSTOMER REQUIREMENTS/PURPOSE	The user is able to post a new advertisement via a form.
PREREQUISITES	<ul style="list-style-type: none"> <li>● A working internet connection</li> <li>● User must be logged in</li> <li>● User opens the advertisement page by selecting the button on the home page.</li> <li>● Then select the plus icon at the right hand bottom side of the page to add a new advertisement.</li> </ul>
STEPS	<ul style="list-style-type: none"> <li>● Click on the advertisement button on the home page.</li> <li>● Select the plus icon at the right hand bottom side of the page to add a new advertisement.</li> <li>● Fill the form and submit.</li> </ul>
EXPECTED RESULT	The app should save the form entry and show the new advertisement in the advertisements page
TEST INPUT DATA	Form entry
OUTPUT	The data is successfully stored in the database and the page showing all the advertisements posted on the app has this advertisement.(Incomplete entries were and submitted)
STATUS	PASS

---

<b>TEST CASE TITLE</b>	<b>NEWS PAGE</b>
<b>CUSTOMER REQUIREMENTS/PURPOSE</b>	The user is able to open the news page.
<b>PREREQUISITES</b>	<ul style="list-style-type: none"> <li>● A working internet connection</li> <li>● User must be logged in</li> <li>● User opens the news page by selecting the button on the home page.</li> </ul>
<b>STEPS</b>	<ul style="list-style-type: none"> <li>● Click on the news button on the home page.</li> <li>● Select any news article to open it.</li> </ul>
<b>EXPECTED RESULT</b>	The app should first display the latest news in the news page and open the whole document in the systems web-browser.
<b>TEST INPUT DATA</b>	None
<b>OUTPUT</b>	Latest news was successfully displayed and the link opened successfully.
<b>STATUS</b>	PASS

---

<b>TEST CASE TITLE</b>	<b>LOGOUT</b>
<b>CUSTOMER REQUIREMENTS/PURPOSE</b>	The user is able to logout.
<b>PREREQUISITES</b>	<ul style="list-style-type: none"><li>● A working internet connection</li><li>● User must be logged in</li></ul>
<b>STEPS</b>	<ul style="list-style-type: none"><li>● Click on the logout button</li></ul>
<b>EXPECTED RESULT</b>	The user should be able to log out from the application.
<b>TEST INPUT DATA</b>	None
<b>OUTPUT</b>	The user is logged out of the application successfully.
<b>STATUS</b>	PASS

---

# Future

There can be a lot of future scope for the application as current version at least. Some of the features like the community can be better integrated with the application after there has been some data collected from the customers.

Some of the other feature can be:

Notes and reminders : Adding a feature to take notes and set reminders will increase the productivity if this application.

Fraud Search : As more and more people start using the application, there is bound to be a group of people who will use the application in an unprofessional way. It will be important to find these people. This part of application might become necessity as the application grows.

Improvement to News : The news is another part of application where a lot of improvement is possible. At present we take news from only one service provider, but with time we can add a lot of providers and can add a lot of features such as ranking the news, specialised news for different communities etc.

Messaging : This can be added to the application so that different users can interact with each other.

---

## Requirement Traceability Matrix

The requirements traceability matrix (RTM) reflects full user and system requirements for a system or part of the system. RTM captures all the requirements and their traceability in one document, and the document must be submitted at the end of the life cycle.

RTM is used to record the relationship of requirements for the design, development, testing, and version of the software requirements for a specific version of the software. Changes to requirements are also recorded and tracked in the RTM. RTM is maintained throughout the life cycle of release and have been reviewed and resource data at the end of the release. It is very useful document to track time, change management and risk management in software development.

The requirements Traceability Matrix, which provides detailed idea about the importance of RTM in SDLC. This document presents the requirements traceability matrix (RTM) traces back to the course allocation system and its requirements, design specifications and test scripts.

The table below displays the RTM for the conditions that have been approved to be included in block and tackle system. The following information is provided for each:

T = Test Cases

U = User Requirement

---

ID	User's Requirements	Forward Traceability
U1	User should Sign up	
U2	User should Log in	
U3	User should open Advertisements page	
U4	User should open News page	
U5	User should post advertisements	
U6	Logout	S4

---

ID	System's requirements	Backward Traceability
T1	Register a User	U1
T2	Log in to the application	U1
T3	Empty fields during login/sign Up	U2
T4	Validation of E-Mail	U3
T5	Validate a user	
T6	Validation of Database	
T7	Advertisement page	
T8	Advertisement form	
T9	News page	
T10	Logout	



---

# Requirement Traceability Matrix

REQUIREMENT IDENTIFIERS	User Requirements	U1	U2	U3	U4	U5	U6
Test cases							
T1		X					
T2			X				
T3		X	X				
T4		X	X				
T5			X				
T6			X				
T7				X			
T8						X	
T9					X		
T10							X