



**VIT**<sup>®</sup>  
**Vellore Institute of Technology**  
(Deemed to be University under section 3 of UGC Act, 1956)

## **CSE3002 – INTERNET AND WEB PROGRAMMING**

### **Project-Review – 2**

Title: Welcome to the Neighbourhood

(A social media web application)

**Faculty: Prof. Jayakumar Sadhasivam**

Neelesh Sharma	19BCE0768
Abuzar Bagewadi	19BCE0773
Shaik Md Irfan	19BCE0780
Daksh Paleria	19BCE0779

### **Abstract:**

Due to the pandemic everyone is restricted to their homes and unable to socialize with the outside world and unable to create new networks, friends. Our platform Welcome to Neighbourhood tries to tackle this problem by developing a full-fledged social media web application which can help the users create their own profiles, make new post and moreover it allows the user to connect, chat, share with each other.

### **Problem Statement:**

Present social media applications today are very vast and the connections are wide spread in way this is a good thing as we can farther our reach, but due to the ongoing pandemic situation we are mostly confined to our homes we can't even go to our neighbour's home and the communication among the community is decreasing. Maintaining a healthy and active relationship with community members is very important as they are very near to us and mostly know about the surroundings, they will be swift and first persons to provide any aid or help in emergency situations.

If we post something on today's social media it will be difficult to reach our nearest people due to worldwide presence and also the information which is only applicable to community will be useless to post for the whole world and also there will be privacy issues.

### **Reason for chosen this topic?**

This website is developed keeping in mind the above-mentioned problems. All the information can be easily accessed and shared among the neighbourhood. The needs and requirements of the end users are also kept in mind while designing this website. This website will enable its users to maintain their friend list and user can update their friend list as well as we want to establish a network among the people residing in their neighbourhood

Thus, our social media website developed in order to facilitate its users to establish network between one-to-many persons and maintain all people's profile. It also helps them to save their time and energy when they want to share some kind of information, views, Ideas etc. to their community without risking to get in physical contact during pandemic.

### **Modules:**

Setting up the MVC file structure

Linking mongoDB using mongoose

Authentication using local method (passport.js)

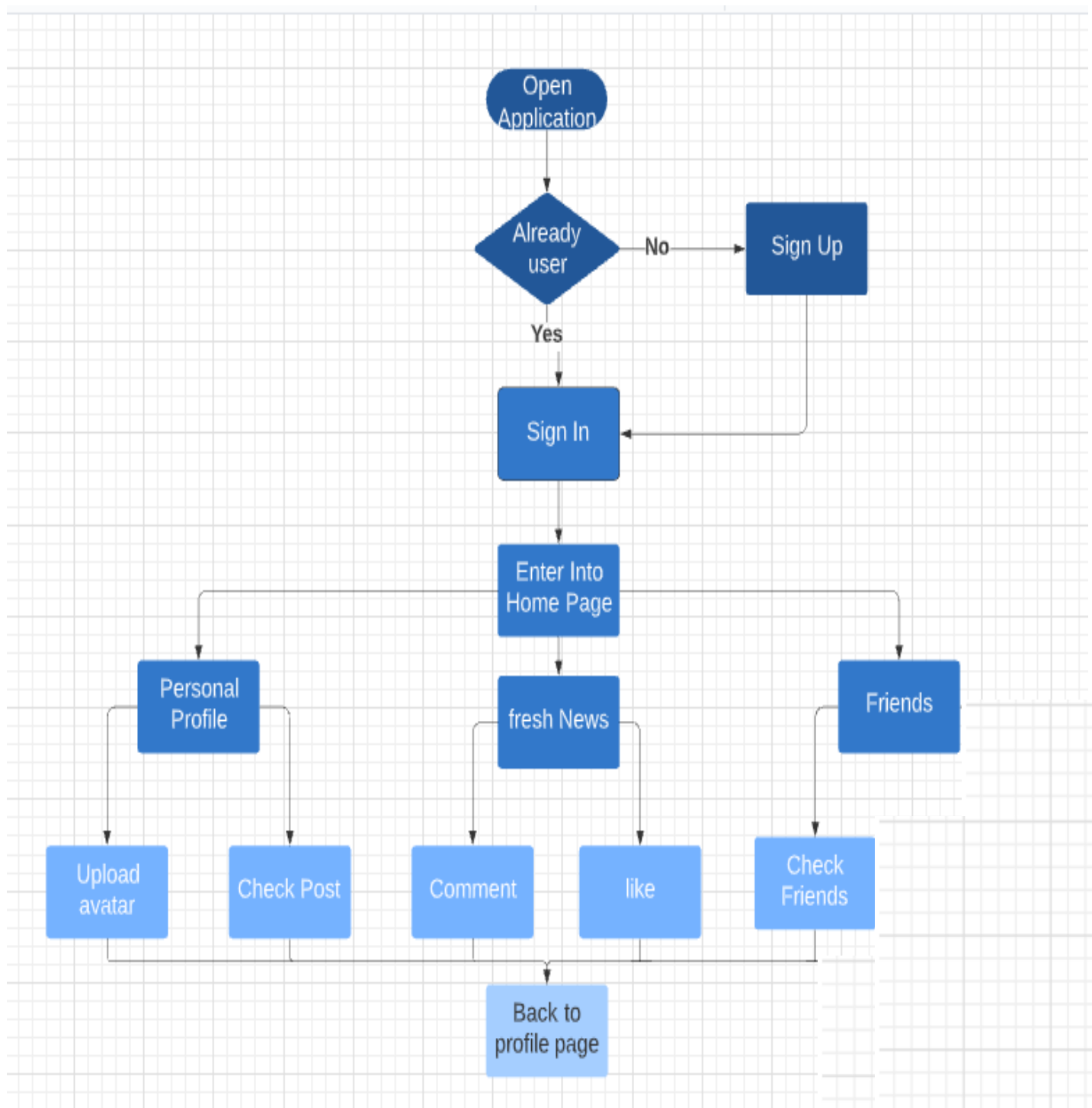
Integrating to passport google auth

Database relations (Posts, Comments)

Setting Friends and Likes

Adding styles

## Process Flow Diagram:



## Prototype Design:

NAV BAR

*Profile Page*


A dark gray rectangular placeholder for a profile page. It features a faint, stylized illustration of a mountain range with a circular sun or moon to its right.

NAV BAR

*News Feed*

A dark gray rectangular placeholder for a news feed. It features a faint, stylized illustration of a mountain range with a circular sun or moon to its right.


Friends and Chat

A dark gray rectangular placeholder for a friends and chat section. It features a faint, stylized illustration of a mountain range with a circular sun or moon to its right.


## Complete Front-end Design

### Sign-In Page

[Don't have account?](#) [SIGN UP](#)





[SIGN IN](#)

 Copyright ©  
Neighbourhood [About Us](#) [Contact Us](#)


### Sign-Up Page

[Already a member?](#) [SIGN IN](#)





[SIGN UP](#)

 Copyright ©  
Neighbourhood [About Us](#) [Contact Us](#)

## Home Page

[SIGN IN](#) [SIGN UP](#)



### Welcome to Neighbourhood

Here you can ask for help from your neighbour community

#### Post your need/reply to the urgent needs of you neighbours

I need a tablet in emergency for fever.  
~ Irfan

I will get you the tablet, let me know its name  
~ Daksh

#### Community members

Irfan

Daksh

 Copyright © Neighbourhood [About Us](#) [Contact Us](#)

## After Sign-In

[SIGN OUT](#)

### Welcome to Neighbourhood

Here you can ask for help from your neighbour community

#### Post your need/reply to the urgent needs of you neighbours

[POST](#)

I need a tablet in emergency for fever. Irfan

[Add comment](#)

~I will get you the tablet, let me know its name. Daksh

#### Community members


Irfan

Daksh


 Copyright © Neighbourhood [About Us](#) [Contact Us](#)

Profile page:

[SIGN OUT](#)



[Choose File](#) [Update](#)

 Copyright ©  
Neighbourhood

[About Us](#) [Contact Us](#)

Database Details:

Storing Users data:

```
const mongoose = require('mongoose');

const multer = require('multer');
const path = require('path');
const AVATAR_PATH = path.join('/uploads/users/avatars')

const userSchema = new mongoose.Schema({
  email: {
    type: String,
    required: true,
    unique: true,
  },
  password: {
    type: String,
    required: true,
  },
  name: {
    type: String,
    required: true,
  },
  avatar: {
    type: String,
  }
},{
  timestamps: true,
});

let storage = multer.diskStorage({
  destination: function(req, file, cb) {
    cb(null, path.join(__dirname, '..', AVATAR_PATH));
  },
  filename: function(req, file, cb) {
    cb(null, file.fieldname + '-' + Date.now());
  }
});

// stsic functions
userSchema.statics.uploadedAvatar = multer({storage:
storage}).single('avatar');
userSchema.statics.avatarPath = AVATAR_PATH;

const User = mongoose.model('User',userSchema);
module.exports = User;
```



storing posts:

```
const mongoose = require('mongoose');
const postSchema = new mongoose.Schema({
  content:{
    type: String,
    required: true
  },
  user:{
    type: mongoose.Schema.Types.ObjectId,
    ref: 'User'
  },
  // include the array of id's of all comments in this post schema itself
  comments: [
    {
      type: mongoose.Schema.Types.ObjectId,
      ref: 'Comment'
    }
  ]
},{
  timestamps: true
});
const Post = mongoose.model('Post',postSchema);
module.exports = Post;
```

Storing Comments:

```
const mongoose = require('mongoose');
const commentSchema = new mongoose.Schema({
  content: {
    type: String,
    required: true
  },
  // comment belongs to the user
  user: {
    type: mongoose.Schema.Types.ObjectId,
    ref: 'User'
  },
  Post: {
    type: mongoose.Schema.Types.ObjectId,
    ref: 'Post'
  }
},{
  timestamps: true
});

const Comment = mongoose.model('Comment',commentSchema);
module.exports = Comment;
```