19CSE314 Software Engineering

Automated Elective Processing

Sample Code

Frontend:

Login page:

```dart
import 'dart:convert';

import 'package:automated_elective_processing/dashboard.dart';
import 'package:automated_elective_processing/functions/notify.dart';
import 'package:automated_elective_processing/models/student.dart';
import 'package:automated_elective_processing/pages/add_user.dart';
import 'package:automated_elective_processing/pages/chooseElective.dart';
import 'package:automated_elective_processing/pages/dashboard/widget/profile_card_widget.dart';
import 'package:automated_elective_processing/pages/dashboard_temp.dart';
import 'package:flutter/material.dart';
import 'package:http/http.dart' as http;

List ele = [];
String src = 'http://localhost:8432';

Run | Debug | Profile
void main() => runApp(const login());

class login extends StatelessWidget {
  const login();

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      debugShowCheckedModeBanner: false,
      routes: {
        '/': (context) => const SignUpScreen(),
        '/welcome': (context) => WelcomeScreen(),
      },
    ); // MaterialApp
  }
}
```

```dart
class SignUpScreen extends StatelessWidget {
  const SignUpScreen();

  @override
  Widget build(BuildContext context) {
    getElectives();
    return Scaffold(
      backgroundColor: Colors.amber,
      body: Column(children: [
        Center(
          child: SizedBox(
            height: 400,
            width: 400,
            child: Image.asset('assets/notification_image.png'),
          ), // SizedBox
        ), // Center
        SizedBox(
          width: 400,
          child: Card(
            child: loginform(),
          ), // Card
        ), // SizedBox
      ]), // Column
    ); // Scaffold
  }
}
```

```dart
class WelcomeScreen extends StatelessWidget {
  const WelcomeScreen();

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      body: Center(
        child: Text('WELCOME!', style: Theme.of(context).textTheme.headline2),
      ), // Center
    ); // Scaffold
  }
}

class loginform extends StatefulWidget {
  const loginform();

  @override
  _loginformState createState() => _loginformState();
}

class _loginformState extends State<loginform> {
  final _username = TextEditingController();
  final _password = TextEditingController();
  double _formProgress = 0;
  final login = GlobalKey<FormState>();
```

```
void _showWelcomeScreen() {
  Navigator.push(
      context,
      MaterialPageRoute(
          builder: (context) =>
              dashboard())); // redirect to dashboard student or faculty // // MaterialPageRoute
}

void _addStudent() {
  Navigator.push(
      context,
      MaterialPageRoute(
          builder: (context) =>
              addUser())); // redirect to dashboard student or faculty // // MaterialPageRoute
}

void _adminDash() {
  Navigator.push(
      context,
      MaterialPageRoute(
          builder: (context) =>
              tempDash())); // redirect to dashboard student or faculty // // MaterialPageRoute
}
```

Student dashboard:

```
import 'package:flutter/material.dart';
import 'package:automated_elective_processing/common/app_colors.dart';

import 'package:provider/provider.dart';

import 'dashboard/dashboard.dart';
import 'widget/side_bar_menu.dart';

class HomePage extends StatefulWidget {
  @override
  _HomePageState createState() => _HomePageState();
}

class _HomePageState extends State<HomePage> {
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      drawer: SideBar(),
      backgroundColor: AppColor.bgSideMenu,
      body: SafeArea(
        child: Row(
          crossAxisAlignment: CrossAxisAlignment.start,
          children: [
              Expanded(
                child: SideBar(),
              ), // Expanded

            /// Main Body Part
            Expanded(
              flex: 4,
              child: Dashboard(),
            ), // Expanded
          ],
        ), // Row
      ), // SafeArea
```

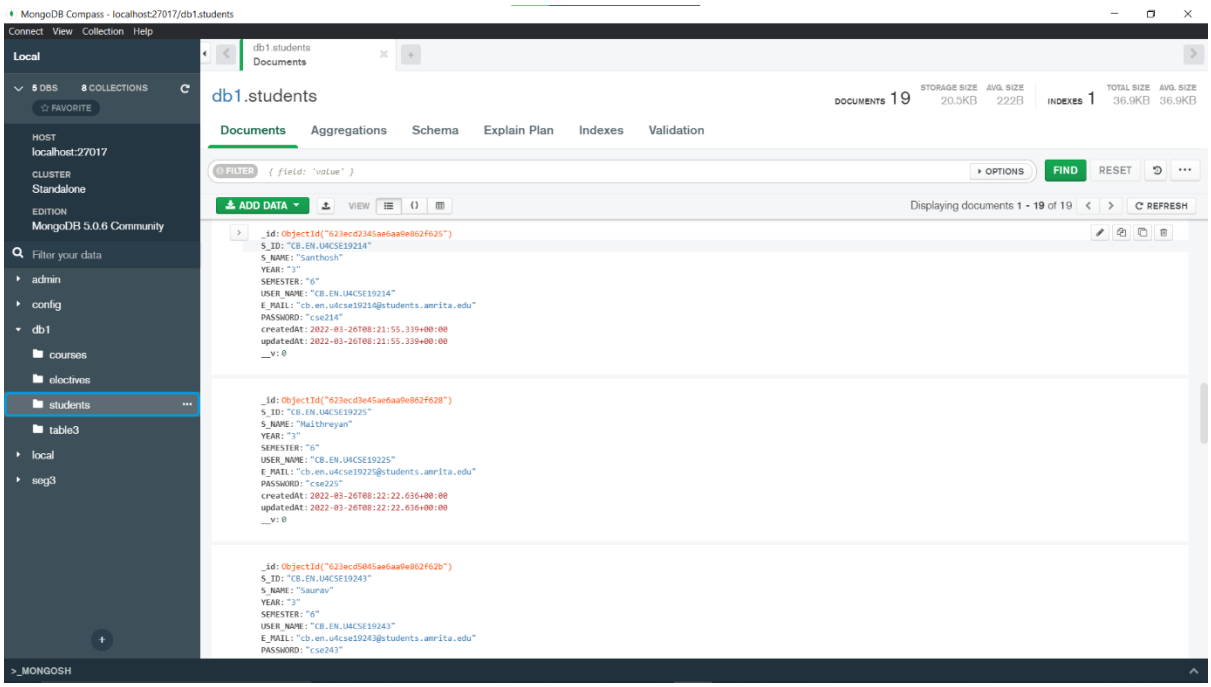
```dart
import 'package:automated_elective_processing/pages/chooseElective.dart';
import 'package:flutter/material.dart';
import 'package:automated_elective_processing/common/app_colors.dart';
import 'package:automated_elective_processing/pages/change_pwd.dart';
import 'package:automated_elective_processing/pages/logout.dart';

import '../WIP.dart';

class SideBar extends StatefulWidget {
  @override
  _SideBarState createState() => _SideBarState();
}

class _SideBarState extends State<SideBar> {
  @override
  Widget build(BuildContext context) {
    return Drawer(
      elevation: 0,
      child: Container(
        color: AppColor.bgSideMenu,
        child: Column(
          crossAxisAlignment: CrossAxisAlignment.start,
          children: [
            Container(
              margin: EdgeInsets.symmetric(horizontal: 20, vertical: 20),
              child: Text(
                "CAUT-Dashboard",
                style: TextStyle(
                  color: AppColor.yellow,
                  fontSize: 25,
                  fontWeight: FontWeight.bold,
                ), // TextStyle
              ), // Text
            ), // Container
```

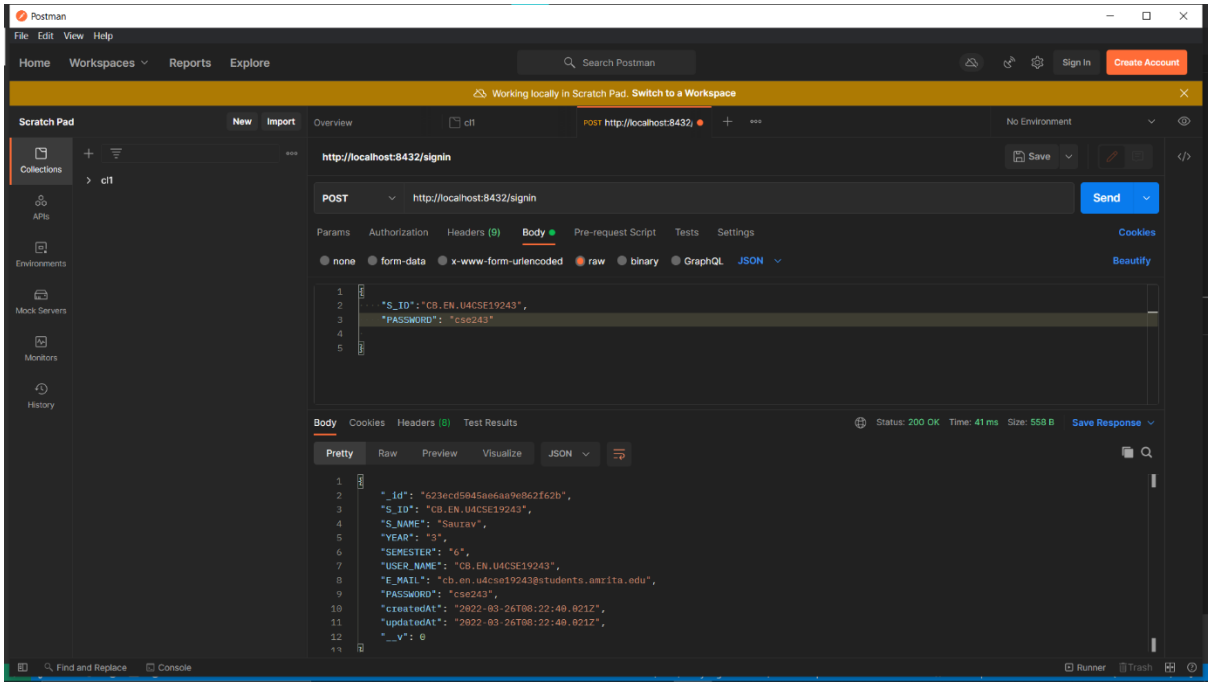
```dart
            DrawerListTile(
              title: "Change Password",
              icon: "assets/reset_password.png",
              press: () {
                Navigator.push(context, MaterialPageRoute(builder: (context) => change_pwd()));
              },
            ), // DrawerListTile
            DrawerListTile(
              title: "Elective Agreement policy",
              icon: "assets/menu_onboarding.png",
              press: () {
                Navigator.push(context, MaterialPageRoute(builder: (context) => WIP()));
              },
            ), // DrawerListTile
            DrawerListTile(
              title: "Elective Preference Selection",
              icon: "assets/menu_report.png",
              press: () {

                Navigator.push(context, MaterialPageRoute(builder: (context) => chooseElective()));
              },
            ), // DrawerListTile
            DrawerListTile(
              title: "Change Elective",
              icon: "assets/menu_onboarding.png",
              press: () {
                Navigator.push(context, MaterialPageRoute(builder: (context) => WIP()));
              },
            ), // DrawerListTile
            DrawerListTile(
              title: "Logout",
              icon: "assets/logout.png",
              press: () {
                Navigator.pushReplacement(context, MaterialPageRoute(builder: (context) => logout()));
              },
```

```dart
import 'dart:html';

import 'package:flutter/material.dart';
import 'package:automated_elective_processing/common/app_colors.dart';
import 'package:intl/intl.dart';
import 'package:table_calendar/table_calendar.dart';

class CalendarWidget extends StatefulWidget {
  @override
  _CalendarWidgetState createState() => _CalendarWidgetState();
}

class _CalendarWidgetState extends State<CalendarWidget> {
  DateTime _focusedDay = DateTime.now();

  @override
  Widget build(BuildContext context) {
    return Container(
      decoration: BoxDecoration(
          color: AppColor.white, borderRadius: BorderRadius.circular(10)), // BoxDecoration
      padding: EdgeInsets.all(10),
      child: Column(
        children: [
          Row(
            mainAxisAlignment: MainAxisAlignment.spaceBetween,
            children: [
              Text(
                "${DateFormat("MMM, yyyy").format(_focusedDay)}",
                style: TextStyle(fontWeight: FontWeight.bold, fontSize: 20),
              ), // Text
              Row(
                children: [
                  InkWell(
                    onTap: () {
                      setState(() {
```

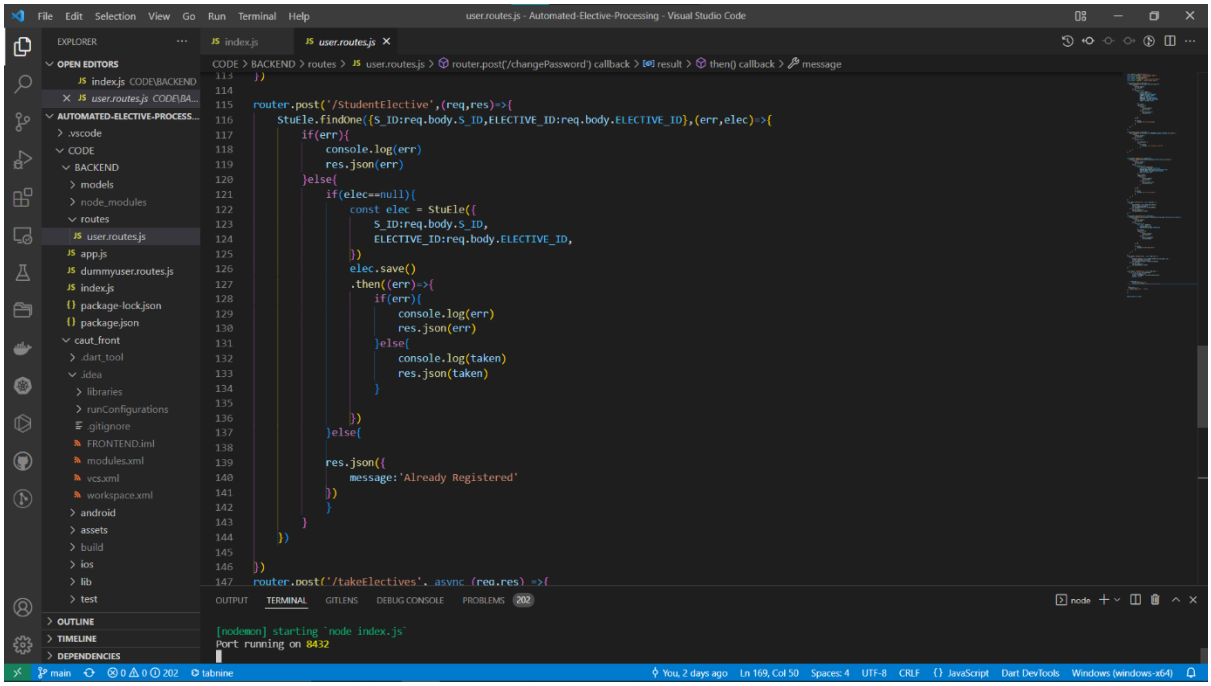
Database

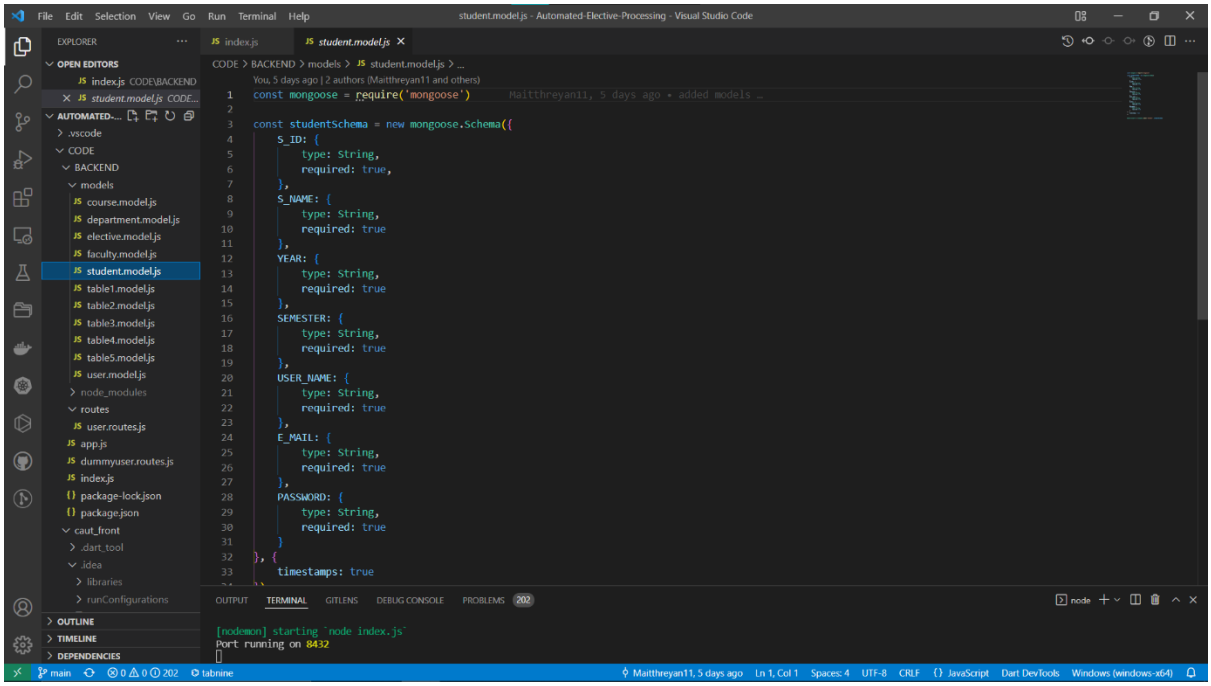
```
(base) PS D:\FILES\OFFICIAL\SEM 6\SE\PROJECT\Automated-Elective-Processing> docker pull mongo:latest
latest: Pulling from library/mongo
Digest: sha256:ad947856db716ddd0b9cc525e341c77208ed8dafcb4a6ad23f9b3addd7a4f71c
Status: Image is up to date for mongo:latest
docker.io/library/mongo:latest
```



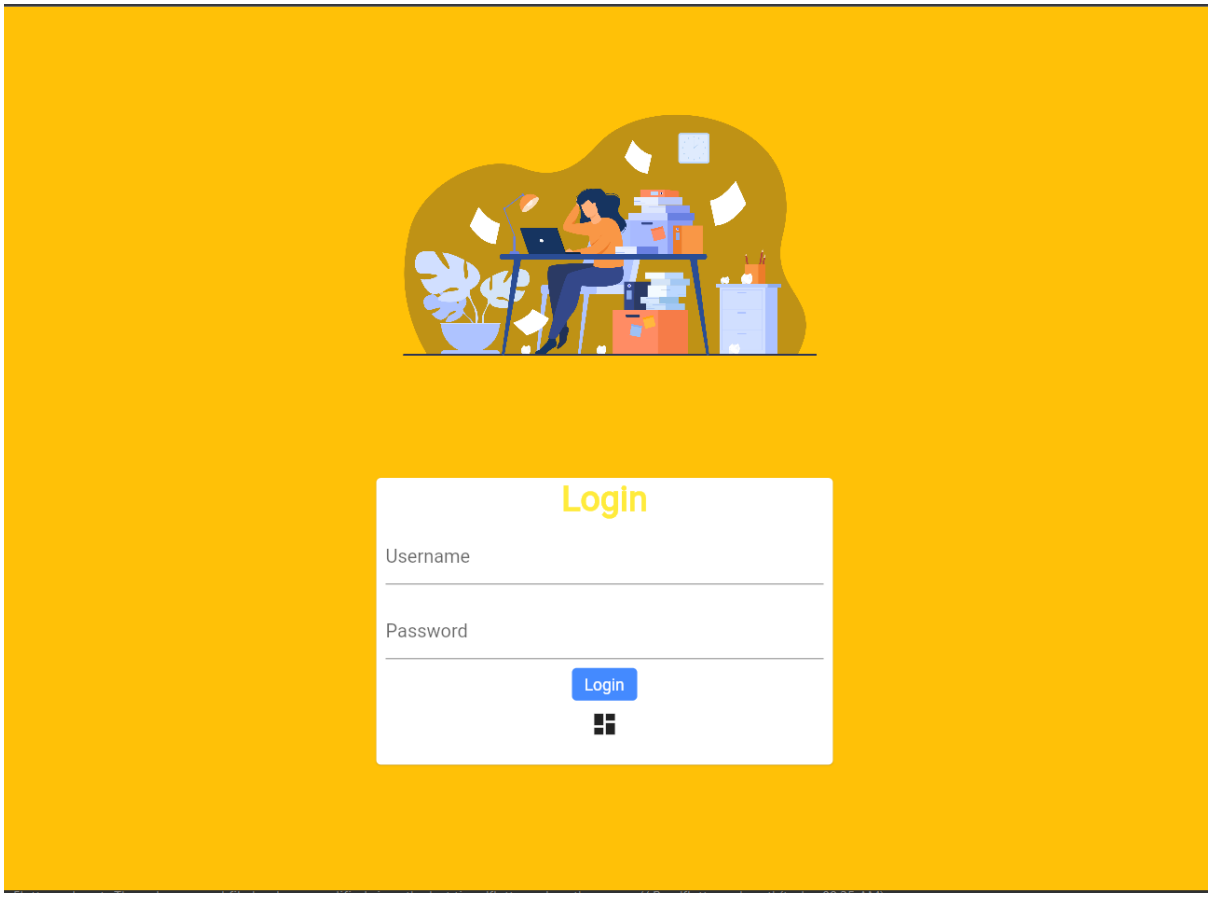

Postman
File Edit View Help
Home Workspaces Reports Explore
Search Postman
Sign In
Create Account
Working locally in Scratch Pad. Switch to a Workspace
Scratch Pad
New Import
Collections
APIs
Environments
Mock Servers
Monitors
History
cl1
Overview
POST http://localhost:8432/
No Environment
http://localhost:8432/signin
Save
POST
http://localhost:8432/signin
Send
Params Authorization Headers (9) Body Pre-request Script Tests Settings
Cookies
none form-data x-www-form-urlencoded raw binary GraphQL JSON
Beautify
{
"S_ID":"CB.EN.U4CSE19243",
"PASSWORD": "cse243"
}
Body Cookies Headers (8) Test Results
Status: 200 OK Time: 41 ms Size: 558 B
Save Response
Pretty Raw Preview Visualize JSON
{
"_id": "623ecd5045ae6aa9e862f62b",
"S_ID": "CB.EN.U4CSE19243",
"S_NAME": "Saurav",
"YEAR": "3",
"SEMESTER": "6",
"USER_NAME": "CB.EN.U4CSE19243",
"E_MAIL": "cb.en.u4cse19243@students.amrita.edu",
"PASSWORD": "cse243",
"createdAt": "2022-03-26T08:22:40.021Z",
"updatedAt": "2022-03-26T08:22:40.021Z",
"__v": 0
}
Find and Replace
Console
Runner
Trash

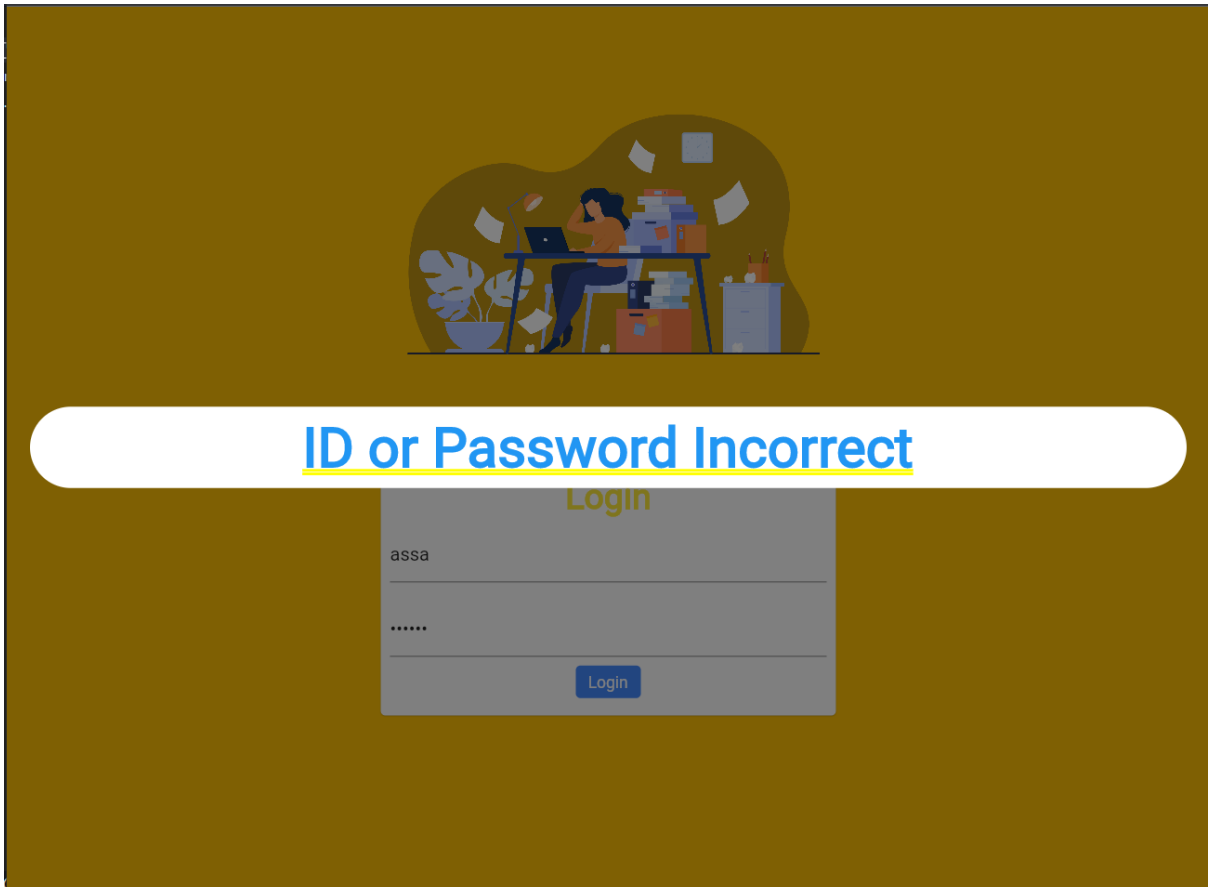
Backend:


File Edit Selection View Go Run Terminal Help
user.routes.js - Automated-Elective-Processing - Visual Studio Code
EXPLORER
OPEN EDITORS
index.js CODE\BACKEND
user.routes.js CODE\BA...
AUTOMATED-ELECTIVE-PROCESS...
.vscode
CODE
BACKEND
models
node_modules
routes
user.routes.js
app.js
dummyuser.routes.js
index.js
package-lock.json
package.json
caut_front
.dart_tool
.idea
libraries
runConfigurations
.gitignore
FRONTEND.iml
modules.xml
vcs.xml
workspace.xml
android
assets
build
ios
lib
test
OUTLINE
TIMELINE
DEPENDENCIES
index.js
user.routes.js
CODE > BACKEND > routes > user.routes.js > router.post('/changePassword') callback > result > then() callback > message
})
router.post('/StudentElective',(req,res)=>{
StuEle.findOne({S_ID:req.body.S_ID,ELECTIVE_ID:req.body.ELECTIVE_ID},(err,elec)=>{
if(err){
console.log(err)
res.json(err)
}else{
if(elec==null){
const elec = StuEle({
S_ID:req.body.S_ID,
ELECTIVE_ID:req.body.ELECTIVE_ID,
})
elec.save()
.then((err)=>{
if(err){
console.log(err)
res.json(err)
}else{
console.log(taken)
res.json(taken)
}
})
}else{
res.json({
message:'Already Registered'
})
}
}
})
})
router.post('/takeElectives', async (req,res) =>{
OUTPUT TERMINAL GITLENS DEBUG CONSOLE PROBLEMS 202
node
[nodemon] starting `node index.js`
Port running on 8432
main
You, 2 days ago Ln 169, Col 50 Spaces: 4 UTF-8 CRLF {} JavaScript Dart DevTools Windows (windows-x64)

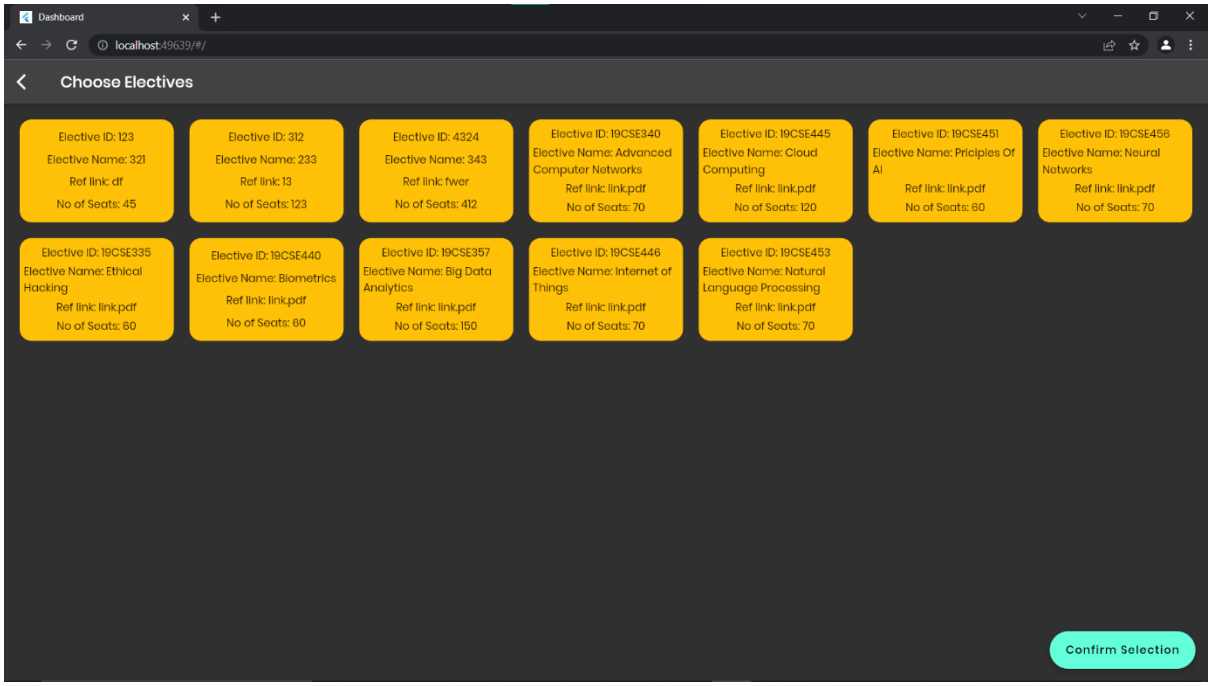


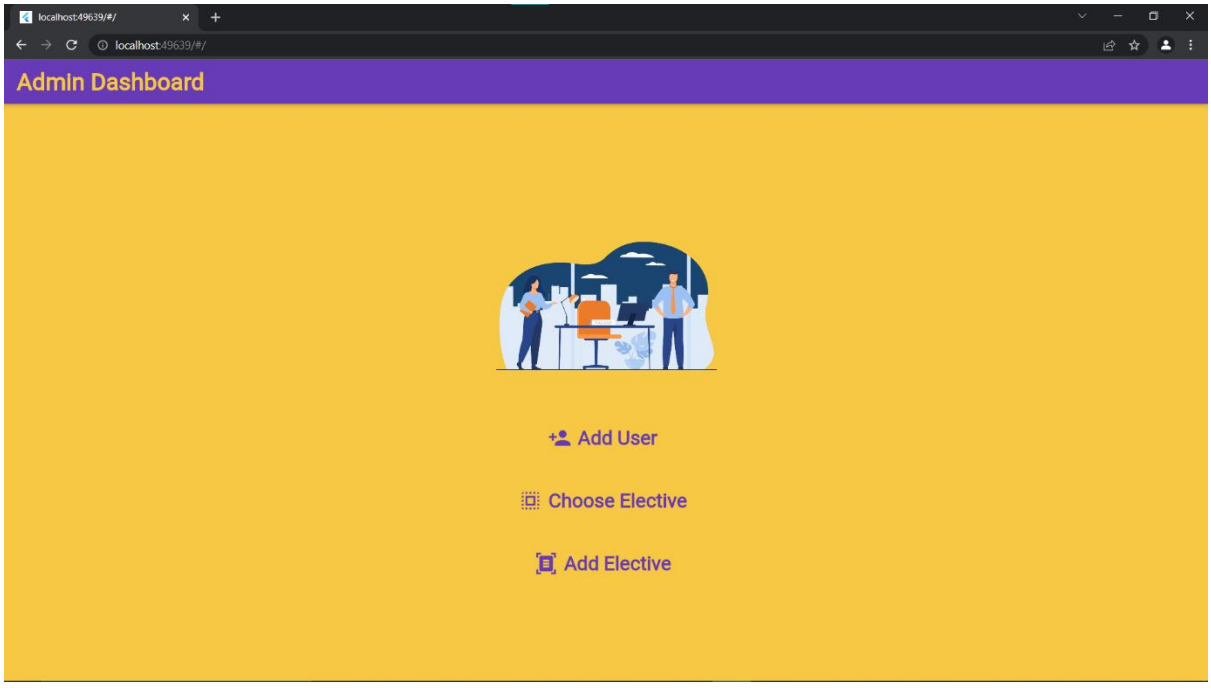
8) Screenshot of the project

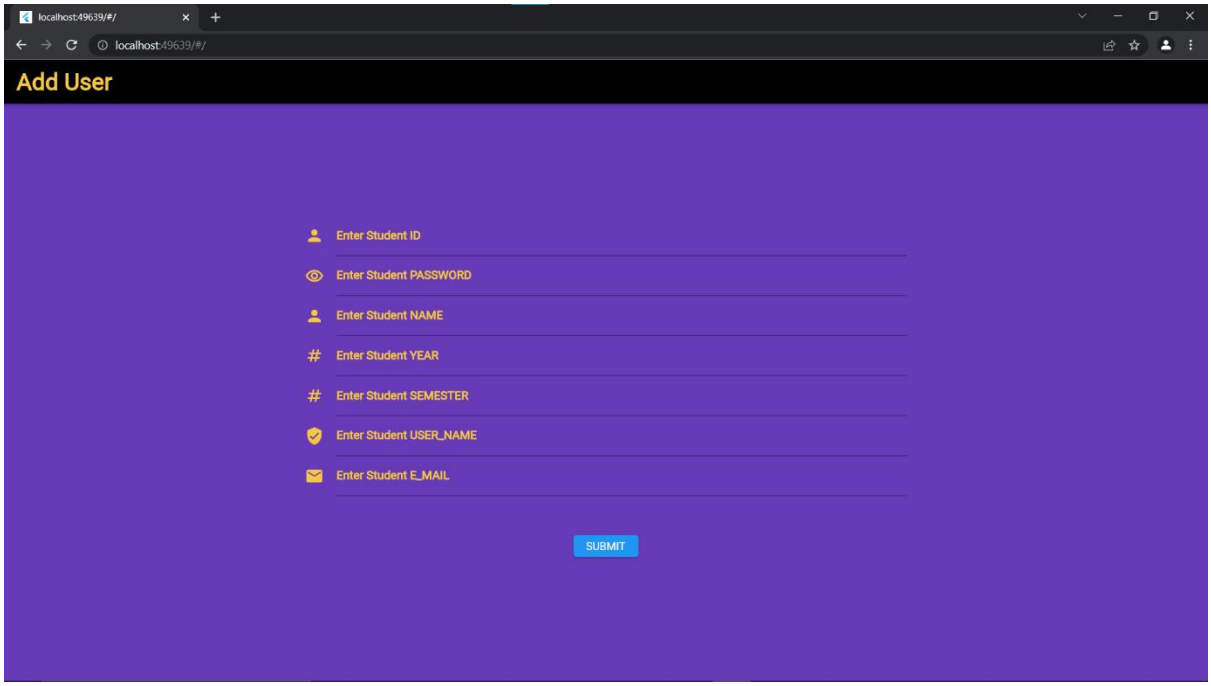


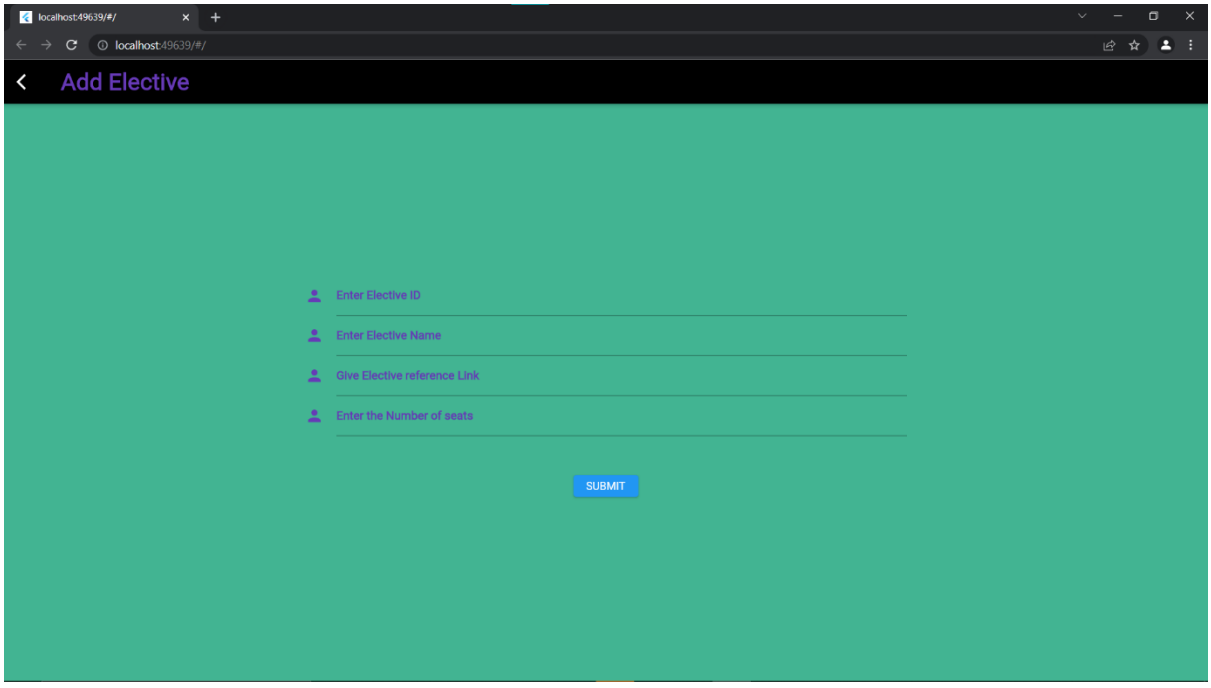

ID or Password Incorrect
Login
assa
.....
Login

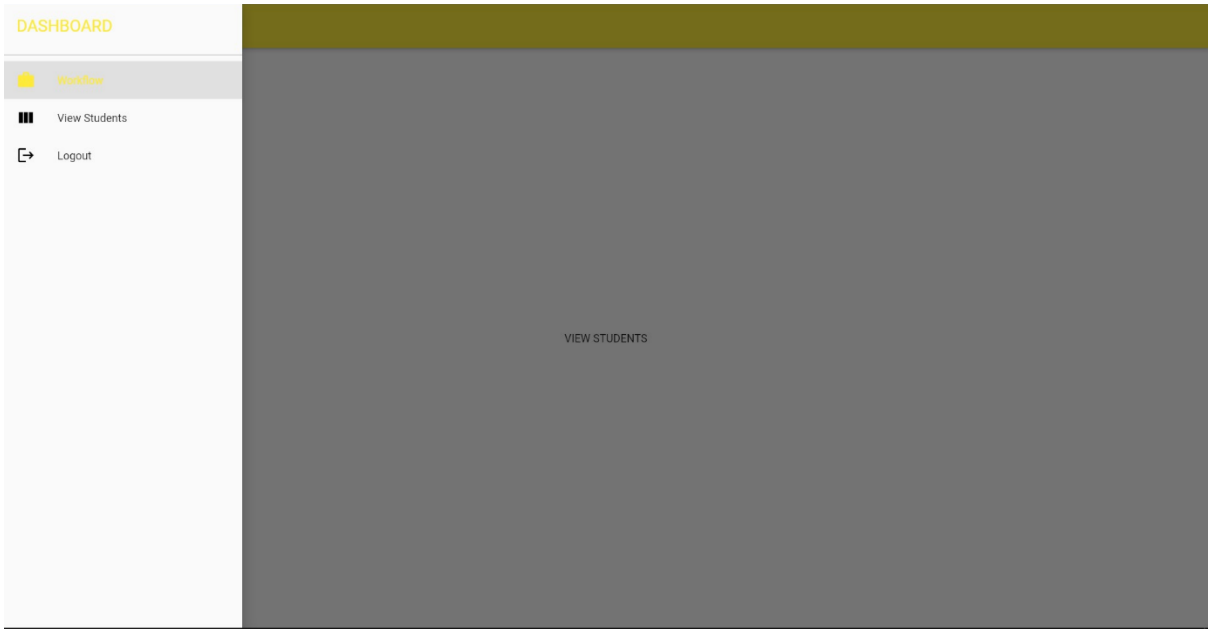
CAUT-Dashboard
Change Password
Elective Agreement policy
Elective Preference Selection
Change Elective
Logout
≡ Dashboard
Welcome tejaswi
Choose your elective preference!
Mar, 2022
SUN MON TUE WED THU FRI SAT
-> The bad news is time flies. The good news is you're the pilot. — Michael Altshuler <-
tejaswi
CB.EN.U4CSE19259
19CSE456
REG

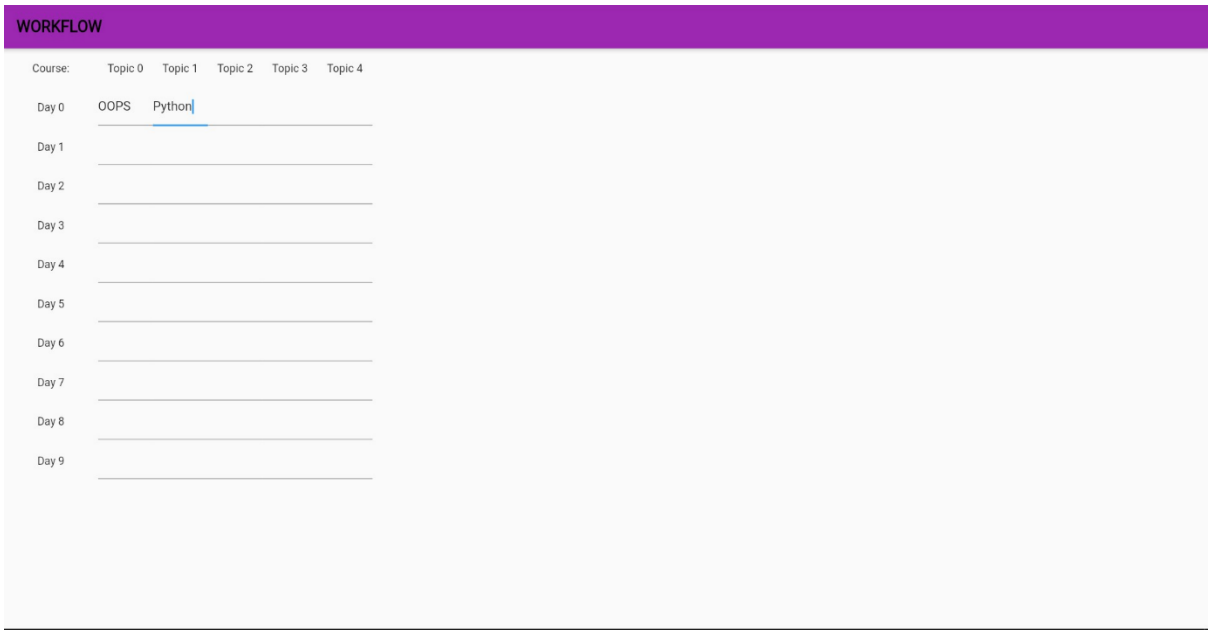

Dashboard
localhost:49639/#/
Choose Electives
Elective ID: 123
Elective Name: 321
Ref link: df
No of Seats: 45
Elective ID: 312
Elective Name: 233
Ref link: 13
No of Seats: 123
Elective ID: 4324
Elective Name: 343
Ref link: fwer
No of Seats: 412
Elective ID: 19CSE340
Elective Name: Advanced Computer Networks
Ref link: link.pdf
No of Seats: 70
Elective ID: 19CSE445
Elective Name: Cloud Computing
Ref link: link.pdf
No of Seats: 120
Elective ID: 19CSE451
Elective Name: Priciples Of AI
Ref link: link.pdf
No of Seats: 60
Elective ID: 19CSE456
Elective Name: Neural Networks
Ref link: link.pdf
No of Seats: 70
Elective ID: 19CSE335
Elective Name: Ethical Hacking
Ref link: link.pdf
No of Seats: 60
Elective ID: 19CSE440
Elective Name: Biometrics
Ref link: link.pdf
No of Seats: 60
Elective ID: 19CSE357
Elective Name: Big Data Analytics
Ref link: link.pdf
No of Seats: 150
Elective ID: 19CSE446
Elective Name: Internet of Things
Ref link: link.pdf
No of Seats: 70
Elective ID: 19CSE453
Elective Name: Natural Language Processing
Ref link: link.pdf
No of Seats: 70
Confirm Selection

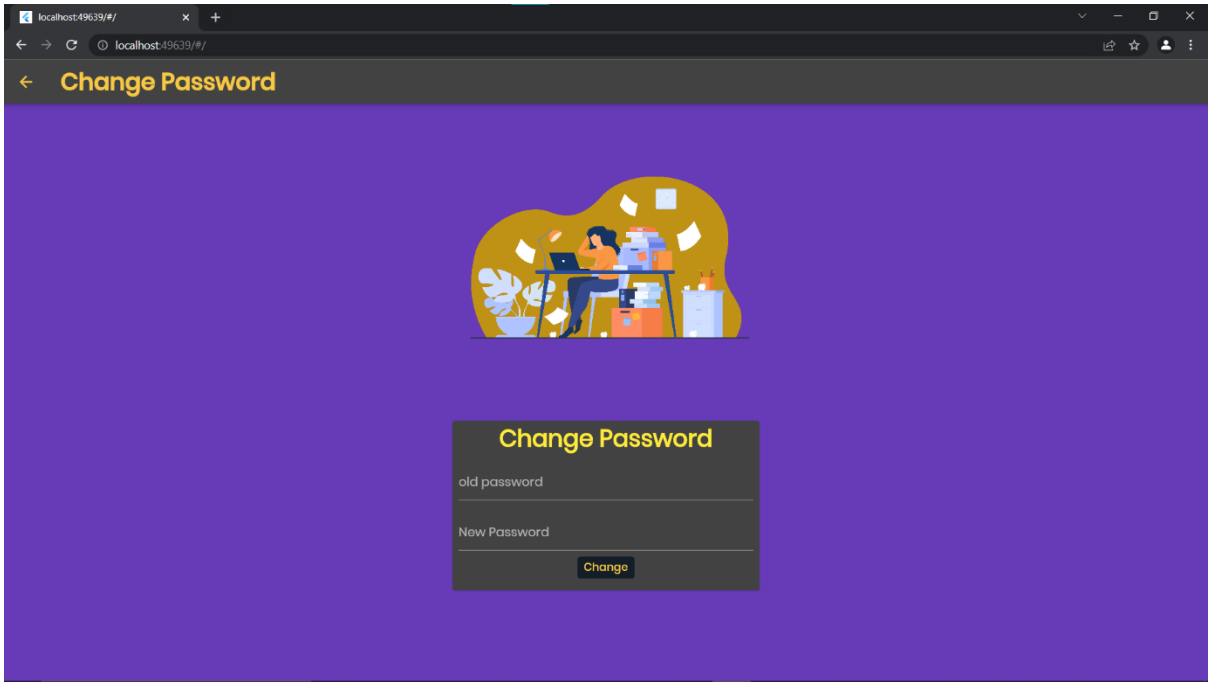

localhost:49639/#/
localhost:49639/#/
Admin Dashboard
Add User
Choose Elective
Add Elective

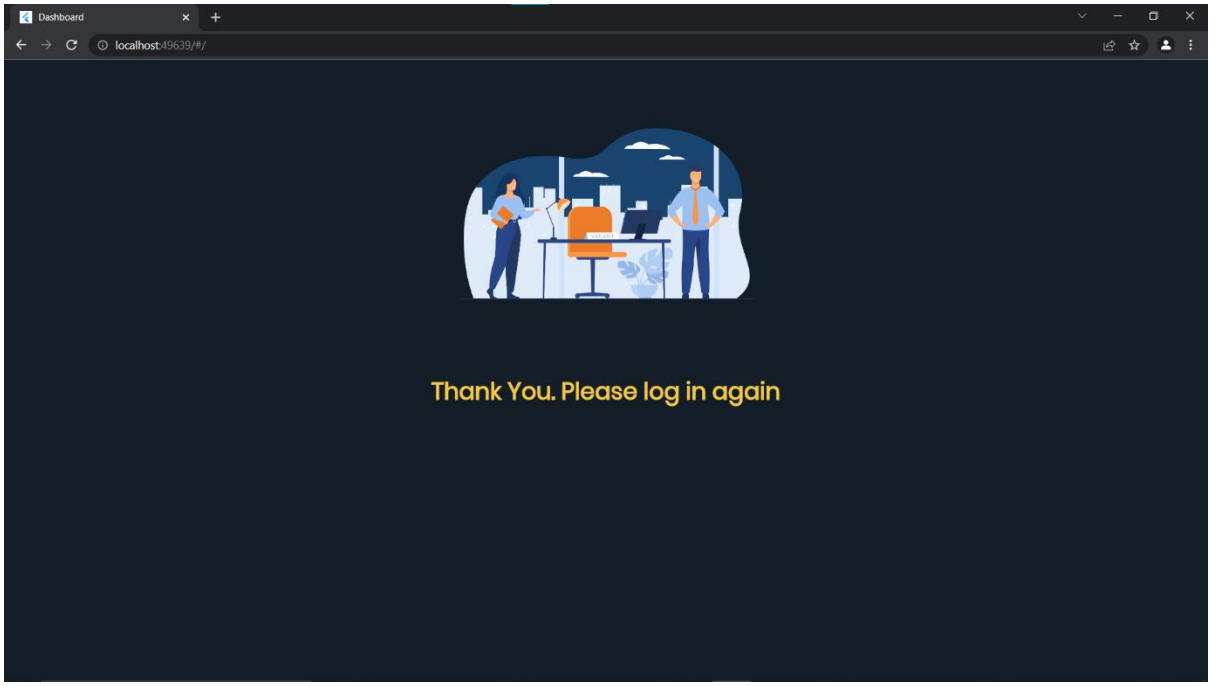

localhost:49639/#/
localhost:49639/#/
Add User
Enter Student ID
Enter Student PASSWORD
Enter Student NAME
Enter Student YEAR
Enter Student SEMESTER
Enter Student USER_NAME
Enter Student E_MAIL
SUBMIT


localhost:49639/#/
localhost:49639/#/
Add Elective
Enter Elective ID
Enter Elective Name
Give Elective reference Link
Enter the Number of seats
SUBMIT


DASHBOARD
Workflow
View Students
Logout
VIEW STUDENTS


WORKFLOW
Course:
Topic 0
Topic 1
Topic 2
Topic 3
Topic 4
Day 0
OOPS
Python
Day 1
Day 2
Day 3
Day 4
Day 5
Day 6
Day 7
Day 8
Day 9


localhost:49639/#/
localhost:49639/#/
Change Password
Change Password
old password
New Password
Change


Dashboard
localhost:49639/#/
Thank You. Please log in again