Software Requirements Specification

for

Circei Education Services

Prepared by,

Akshay Raichur(2GI17CS012)

Akshata Kulkarni(2GI17CS009)

Jayanth Apagundi (2GI17CS182)

KLS GIT, Belagavi

Table of Contents ii

Revision History ii

1. Introduction 1

1.1 Purpose 1

1.2 Definitions, Acronyms and Abbreviations 3

1.3 Intended Audience 4

1.4 Product Scope 4

1.5 Technologies Used 5

1.6 References 6

2. Overall Description 7

2.1 Use Case Model Survey 7

2.2 Entity-Relation Modelling 8

2.3 Product Perspective 8

2.4 Product Functions 8

2.5 User Classes and Characteristics 9

2.6 Sequence Diagram 10

2.7 State Diagram 12

2.8 Design and Implementation Constraints 13

3. External Interface Requirements 13

3.1 User Interfaces 13

3.2 Hardware Interfaces 22

3.3 Software Interfaces 22

3.4 Communications Interfaces 22

6. Conclusion 22

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
| Akshay Raichur  Jayanth Apagundi  (Website) | 24-05-2020 | Initial Draft | 0.1 |
| Akshata Kulkarni  (Mobile App.) | 24-05-2020 | Initial Draft | 0.1 |

1. **INTRODUCTION**

**1.1 Purpose**

The students are important and the future of India. Nowadays, Students play an important role in all the lives and surroundings, hence it is important to educate them so that they can be better and excel the in what they learn and we offer valuable Courses, Workshops, and Conferences which will help the students and increase their confidence in contributing for the well being in the world.

**1.2 Definitions, Acronyms and Abbreviations**

**(Website & Mobile Application)**

*MERN*: MongoDB, Express, React, Node Js

*MongoDB***:** MongoDB is an object-oriented, simple, dynamic, and scalable NoSQL database. It is based on the NoSQL document store model

*Express***:** Express is a module framework for Node that you can use for applications that are based on servers that will listen for any input/connection requests from clients.

*React:* ReactJs is an open-source JavaScript library that is used for building user interfaces specifically for single-page applications.

*Node Js***:** NodeJs is a platform built on Chrome's JavaScript runtime for easily building fast and scalable network applications

*Mongoose:* Mongoose is an Object Data Modeling (ODM) library for MongoDB and NodeJS

*Axios:* Axios is a Javascript library used to make HTTP requests from [node.js](https://nodejs.org/en/) or XMLHttpRequests from the browser.

*React-Routerdom:*React router is dynamic, client-side routing, allows us to build a single-page web application with navigation without the page refreshing as the user navigates. React Router uses component structure to call components, which display the appropriate information.

*HTML:* Hyper Text Markup Language

*Flutter:* Flutter is an open-source UI software development kit created by Google. It is used to develop applications for Android, iOS, Windows, Mac, Linux, Google Fuchsia and the web.

*Rest API:* Representational state transfer is a software architectural style that defines a set of constraints to be used for creating Web services. Web services that conform to the REST architectural style, called RESTful Web services, provide interoperability between computer systems on the Internet.

**1.3 Intended audience and Reading suggestions**

This document is intended for developers, project managers, students, lecturers, users, testers, and documentation writers.

* Developers: This document contains all the architectural diagrams that are helpful for developers building or maintaining similar projects (Educational websites/apps).
* Project managers: This is a detailed document and hence very useful for the ones managing the projects as it contains the flow diagrams and also the database diagrams. This helps the handlers of educational websites/apps understand the project even better.
* Students: The scope of this project in mentioned in this document along with the use case diagram specifying what this project has to offer to a particular audience.
* Lecturers (Admin): Knowing the flow of the application can help the users get a better experience and also explore the product’s capabilities and explore or suggests new ideas.

**1.4 Product Scope**

1. This Organization helps the students to gain immense amount of knowledge and excel in their career.

2. It would show the students the most productive way to achieve their intended goals.

3. It would not only be helpful to students, this would also help the enthusiastic persons who are interested in profiting the knowledge can make use of it.

**1.5 Technologies used**

***Mern stack (Website)***

MERN Stack is a Javascript Stack that is used for easier and faster deployment of full-stack web applications. MERN Stack comprises 4 technologies namely: [MongoDB](https://www.geeksforgeeks.org/mongodb-an-introduction/), [Express](https://www.geeksforgeeks.org/introduction-to-express/), [React](https://www.geeksforgeeks.org/react-js-introduction-working/) and [Node.js.](https://www.geeksforgeeks.org/introduction-to-nodejs/) It is designed to make the development process smoother and easier.Each of these 4 powerful technologies provides an end-to-end framework for the developers to work in and each of these technologies play a big part in the development of web applications.

**A. Mongodb**

Mongodb is an open-source, document-based database. It is also called a noSQL database program. It makes use of JSON like documents along with schemas. Because of the tremendous increase in the volume and variety of data over the past few years, there has been a need for a non-relational database. Mongodb fulfills these criteria, as it stores its data in a document-oriented, file structure. Each database can have multiple collections and data in those collections is stored in the form of an array of JSON objects.

**B. Express.js**

Express.js is a minimal and flexible Node.js web application framework which provides a robust set of fundamental web application features without obscuring Node.js features. Express makes it easy to create APIs because of access to middleware. Middleware functions are functions which have access to the request and response objects along with a next function, which when invoked, executes the middleware succeeding the current middleware. Express.js provides a similar functionality to that of what spring offers to Java applications, that is, an easy to use web framework.

**C. React**

React is a declarative efficient and flexible front end JavaScript library for user interface development. React Js can be used for both client and server side. React anchors the MERN stack. In some sense, this is the defining component of the MERN stack. React is an open-source JavaScript library maintained by Facebook that can be used for creating views rendered in HTML. Unlike AngularJS, React is not a framework. It is a library.

**D. Node.js**

Node.js is an event-driven asynchronous JavaScript runtime which was primarily designed to make scalable network applications and APIs. Node Js is the best choice for real time and streaming applications. Node Js was built on Chrome V8 Engine which is an open-source JavaScript engine developed on C++ and V8 implements the ECMA Script. Although the Node is mentioned as the last tool of the MERN stack, it is the most important one. Node.js' package ecosystem, npm (node package manager), is the largest ecosystem of open source libraries in the world which provides all the modules and dev Dependencies for Angular CLI along with all the Dependencies for Express. It acts as the dependency injector of the stack.

***Flutter (Mobile Application)***

Flutter is an app SDK for building high-performance, high-fidelity apps for iOS, Android, web, and desktop from a single codebase. It enables the developers to deliver high-performance apps that feel natural on different platforms. We embrace differences in scrolling behaviors, typography, icons, and more. It is highly productive; create beautiful, highly-customized user experiences. Flutter web apps can run on any web server. We can run flutter web apps on node.js server and it is incredibly good at servicing large volumes of simple requests, and can code the frontend and backend in javascript and so on.

**1.6 References**

* <https://medium.com/@beaucarnes/learn-the-mern-stack-by-building-an-exercise-tracker-mern-tutorial-59c13c1237a1>
* <https://flutter.dev/docs/resources/technical-overview>
* <https://blog.logrocket.com/flutter-web-app-node-js/>

1. **OVERALL DESCRIPTION**

**2.1 Use case model survey**

****

*Actor Name: User (Ex. Students)*

The user/students has to register himself/herself and login into the website/mobile application where in he/she will be having a user profile and the user has to enroll for available courses/workshops/conferences and pay the respective fees.

Use cases:

1. Registration
2. Login
3. User Profile
4. Enrollment
5. Payment

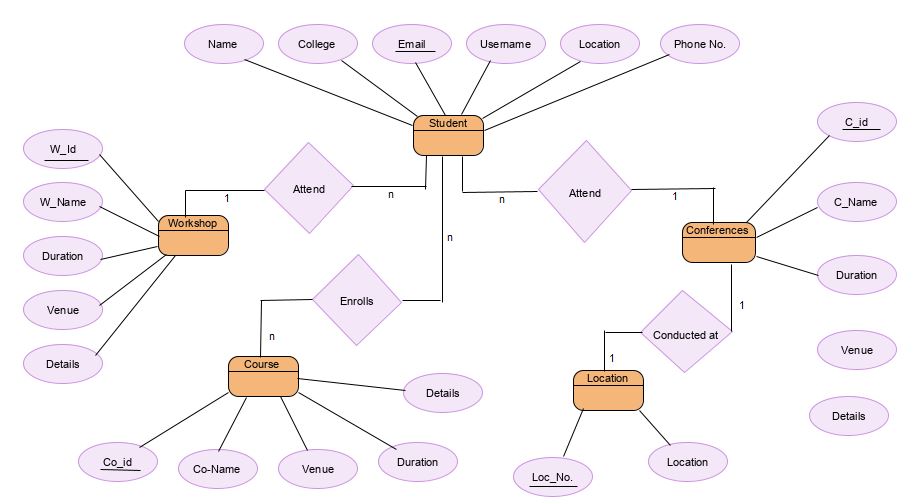
*Actor Name: Admin*

Here, Admin can lecturer’s or a person who is responsible and handling the respective course/workshop/conferences wherein he can add the courses/workshops/conferences held and send the information to the users whosoever have enrolled to the particular courses/workshops/conferences via. Mail or SMS.

Use cases:

1. Registration
2. Login
3. Add Course/workshop/conference
4. Email/SMS

**2.2 Entity-relations modelling**

****

**2.3 Product perspective**

This project implements a simple website/mobile application wherein the students can educate themselves in a better way and beware of the opportunities. The admin can add the course/workshop/conference and mention the details about it and students can enroll for it by paying the mentioned fee to the respective course/workshop/conferences.

**2.4 Product functions**

The website/mobile application will perform the following tasks:

* *Allow users to register/login as students and admin with special credentials.*

New users can register and already existing users have to login to be able to access the website/mobile application. Data validation takes place before the user in entered into the database.

* *Allow admin to make the entries in database regarding course/workshop/conference*

The admin that is the person in charge of particular course/workshop/conference can add the information and details related to whatever he/she is going to conduct i.e. conference name, venue, date, timings, details and so on.

* *The admin can modify or rectify the course/workshop/conference*

He/she can modify or rectify the details of course/workshop/conference at any point of time so that the students can get the information and updates about it.

* *Allow students to go through the available courses/workshops/conferences*

Students can go through the available courses/workshops/conferences and get to know the details regarding them like syllabus, lecturer handling, duration and so on.

* *Students will have their own user profile*

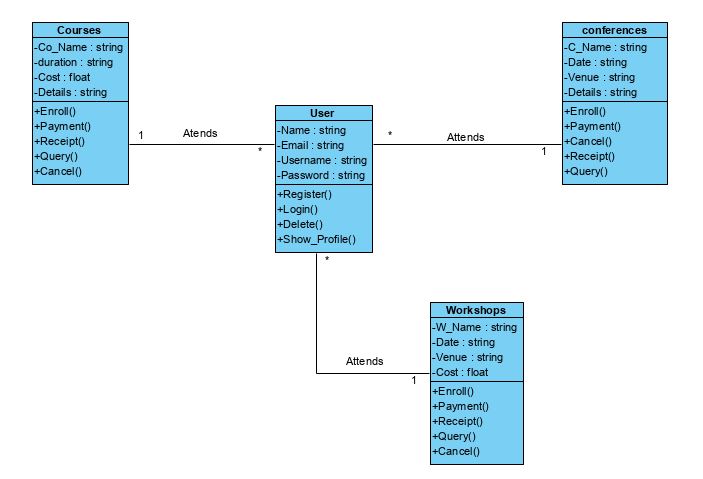
As students register in our website/mobile application they will have their user profile where in they can track their activities, edit their information, see the course/workshop/conferences they have enrolled.

* *Students can enroll for multiple number of courses/workshop/conferences*

Students can enroll for any courses/workshops/conferences and there is no restriction of enrolling to particular number of courses.

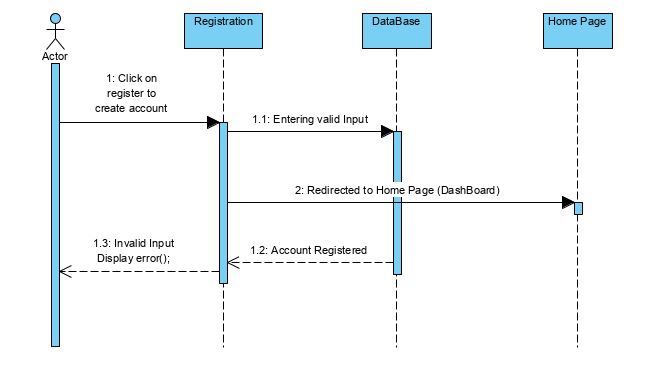
**2.5 User classes and characteristics**

Class diagram:

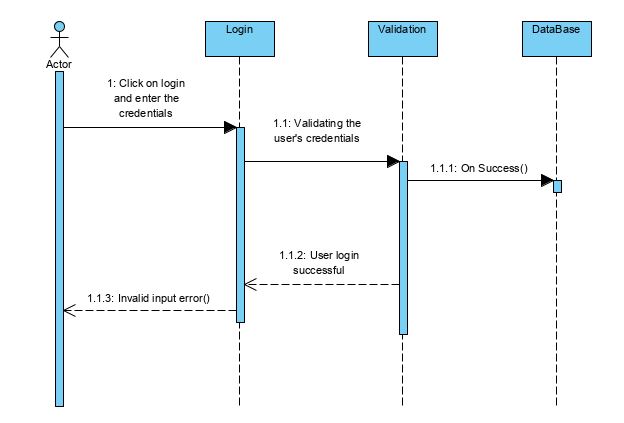


**2.6 Sequence diagram**

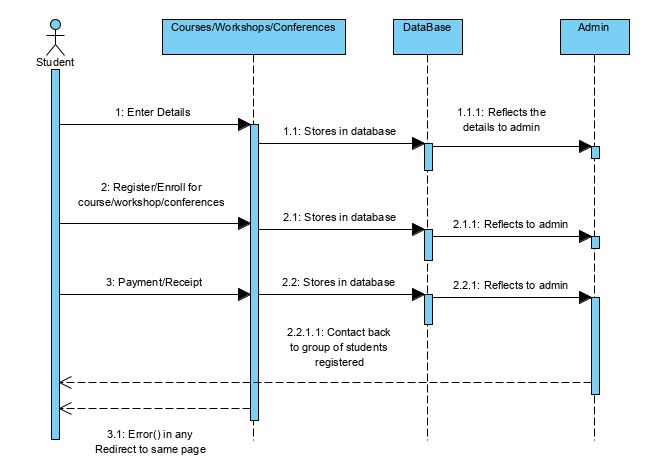
Registration:



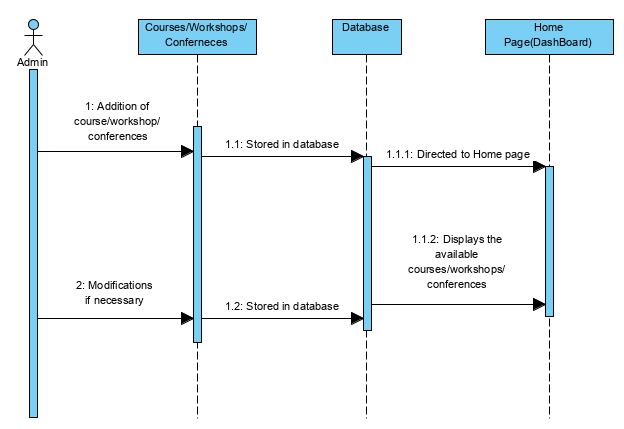
Login



Student

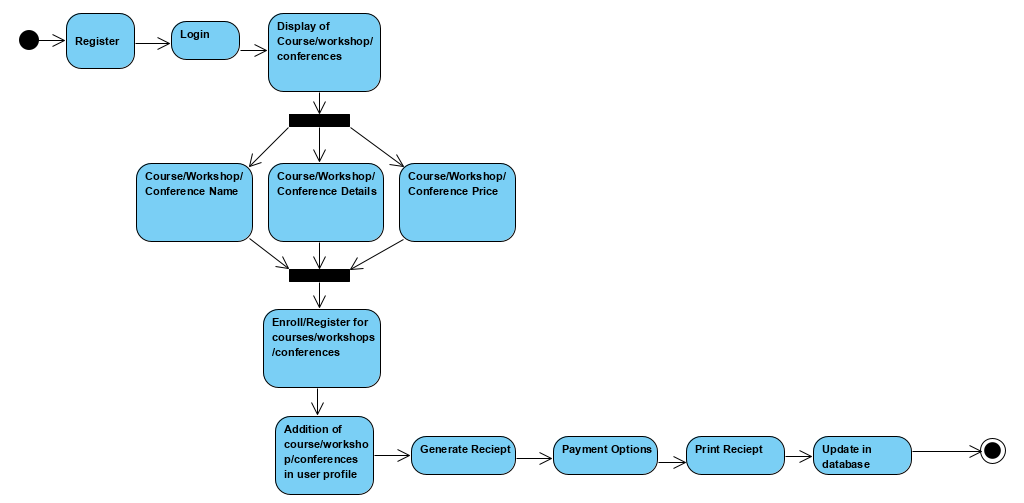


Admin

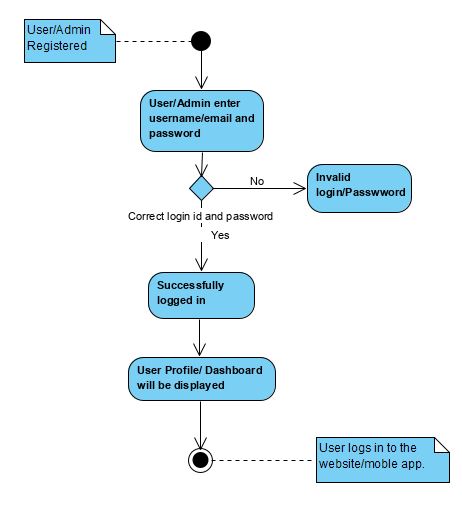


**2.7 State diagram**

Activity modeling



Login activity modelling



**2.8 Design and implementation constraints**

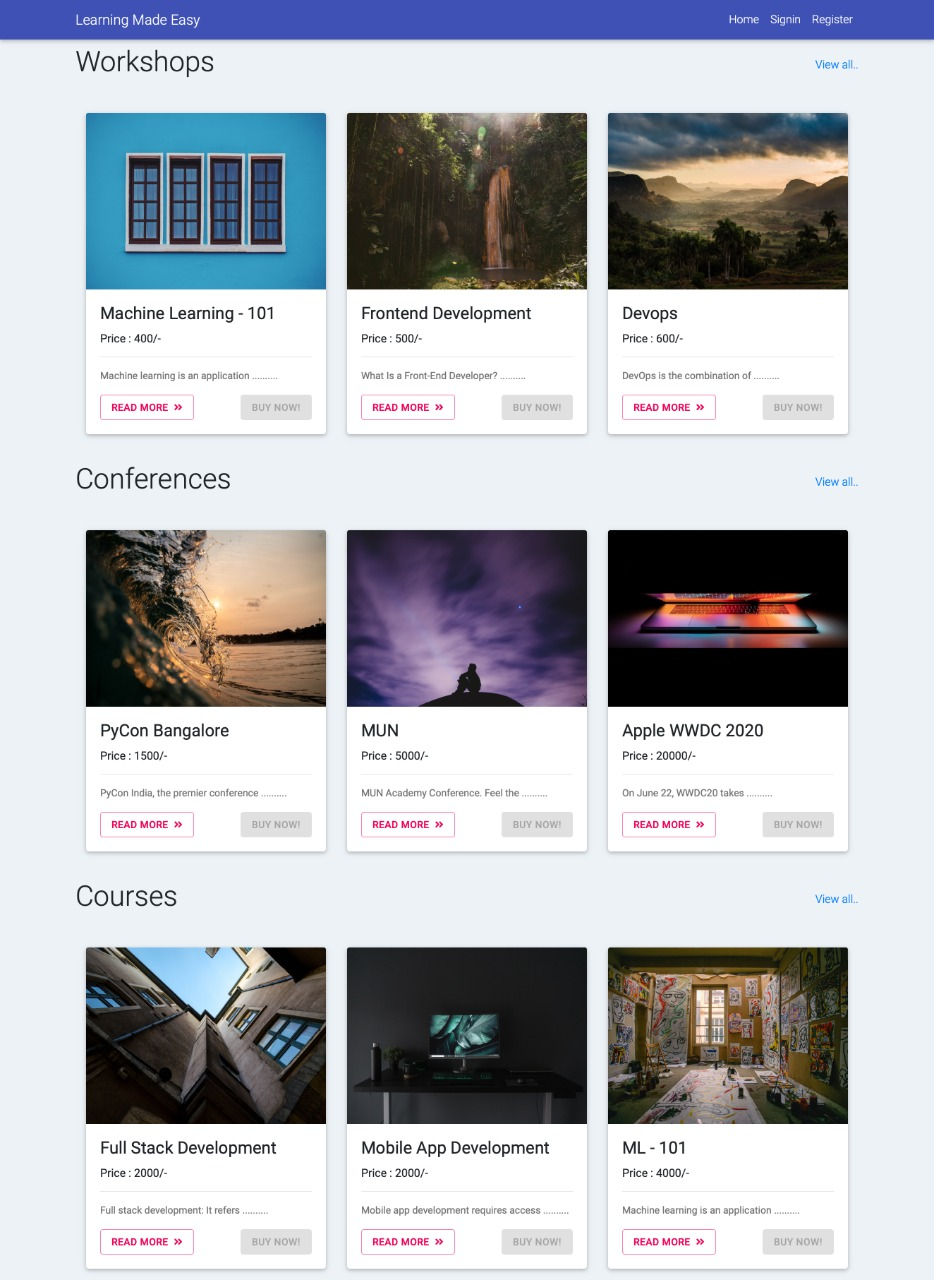
The website/mobile application is developed for an organization for betterment of the students and makes their career gain immense amount of knowledge and admin have different views wherein they can add course/workshop/conference. The payment gateway usually a part of paying fee for enrolled course/workshop/conference has not been implemented and is one of the major constraints. We also need to certify the students after the successful completion of the course/workshop/conference.

1. **EXTERNAL INTERFACE REQUIREMENT**

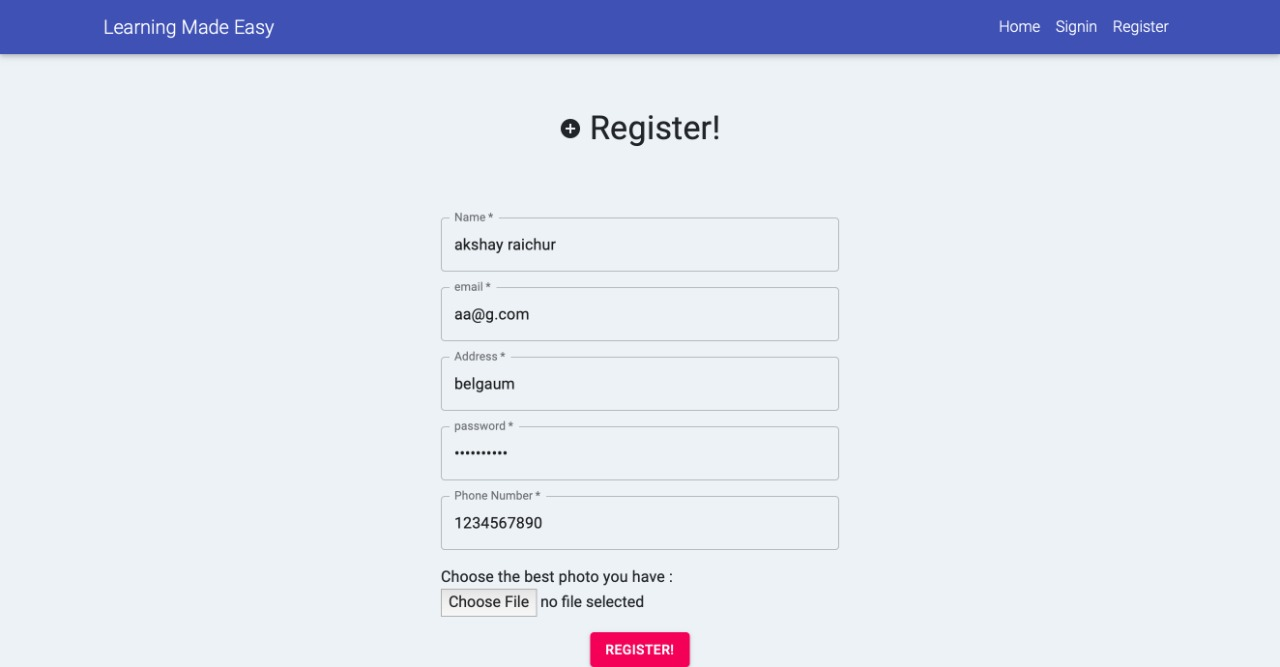
**3.1 User interfaces**

**Web:**

Home Page/ Dashboard:



Registration:



Login:



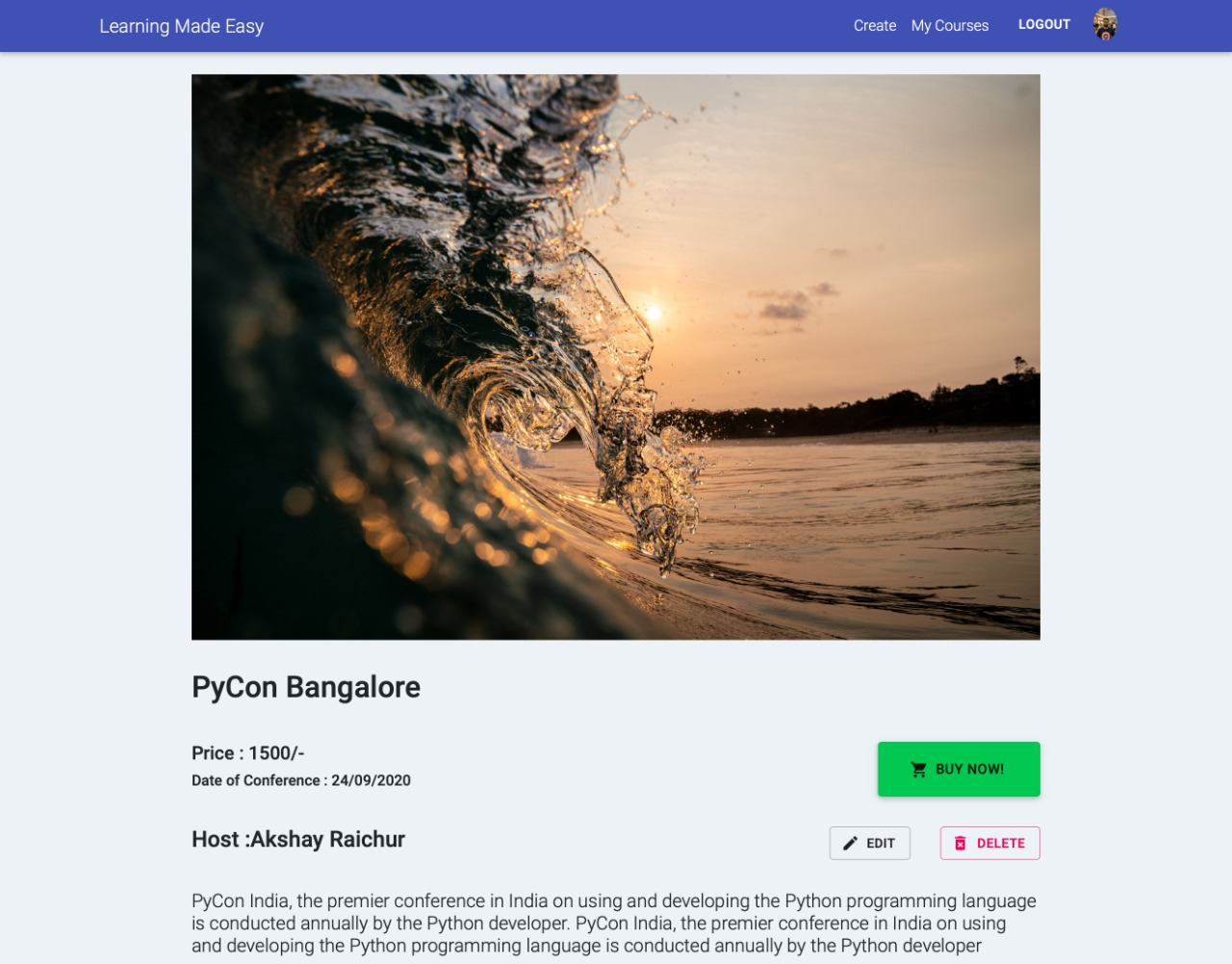
User Profile:



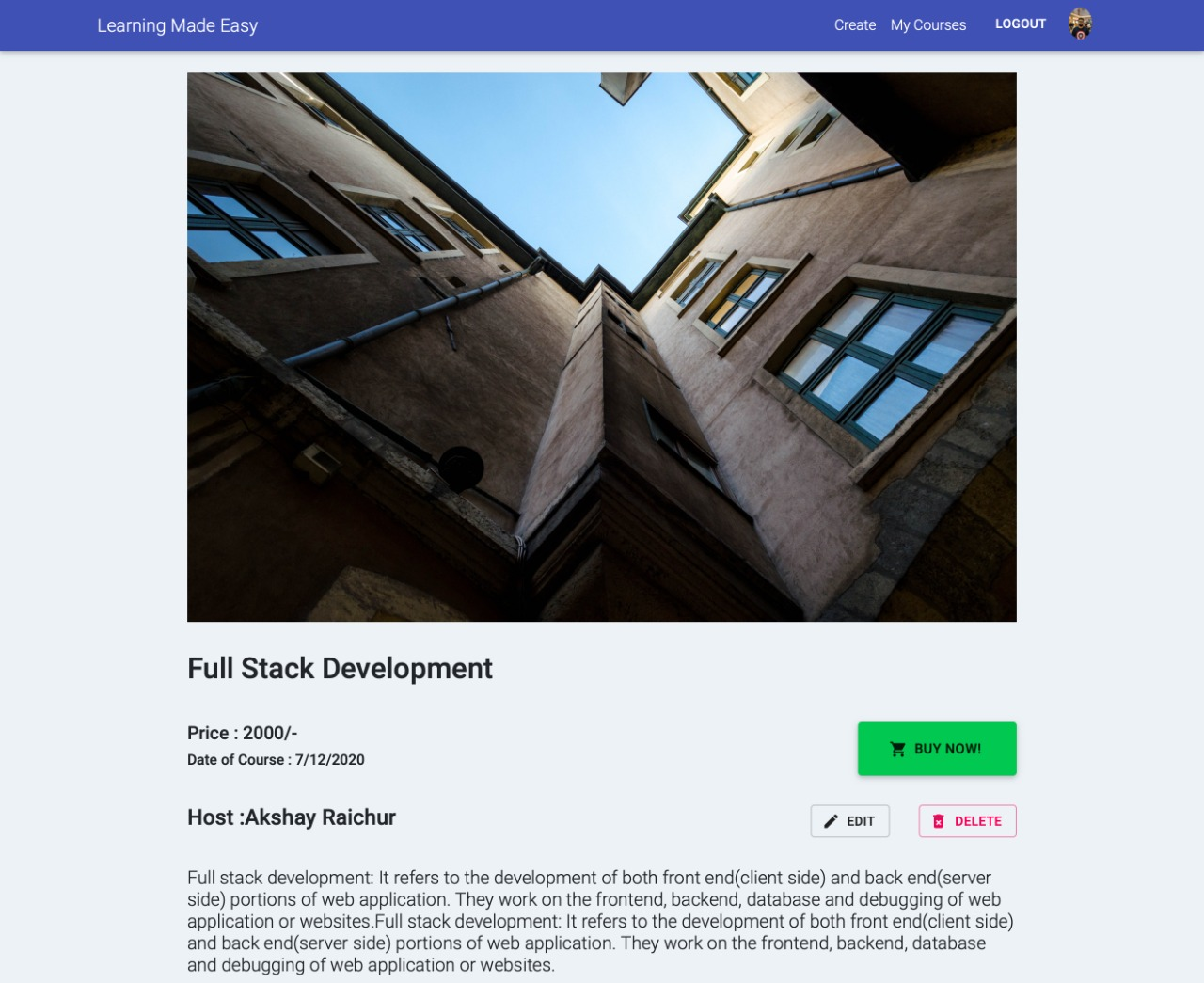
Workshop Details:



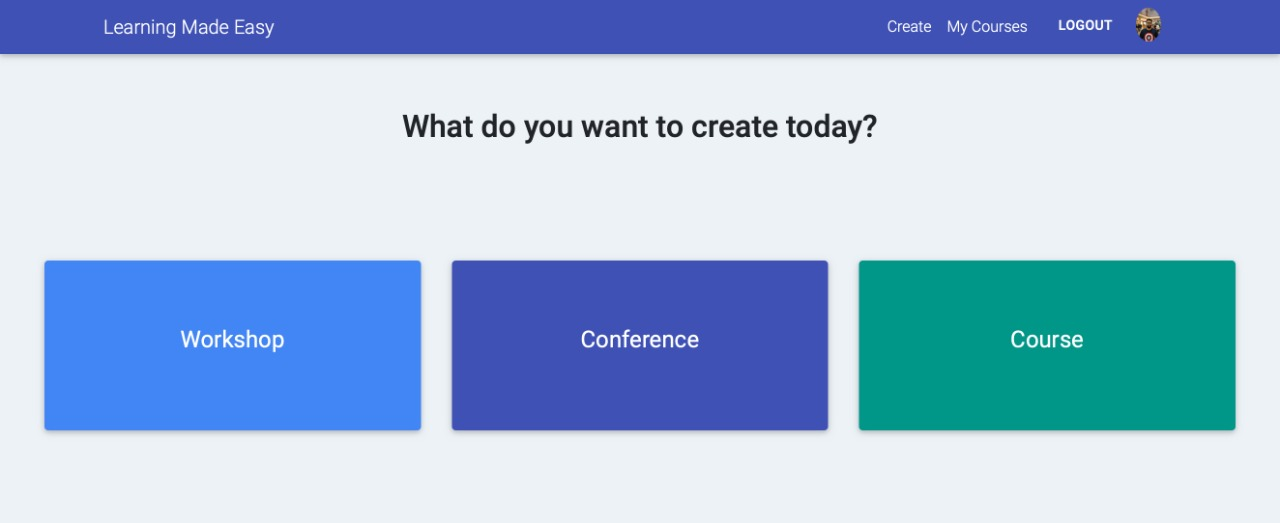
Conference Details:



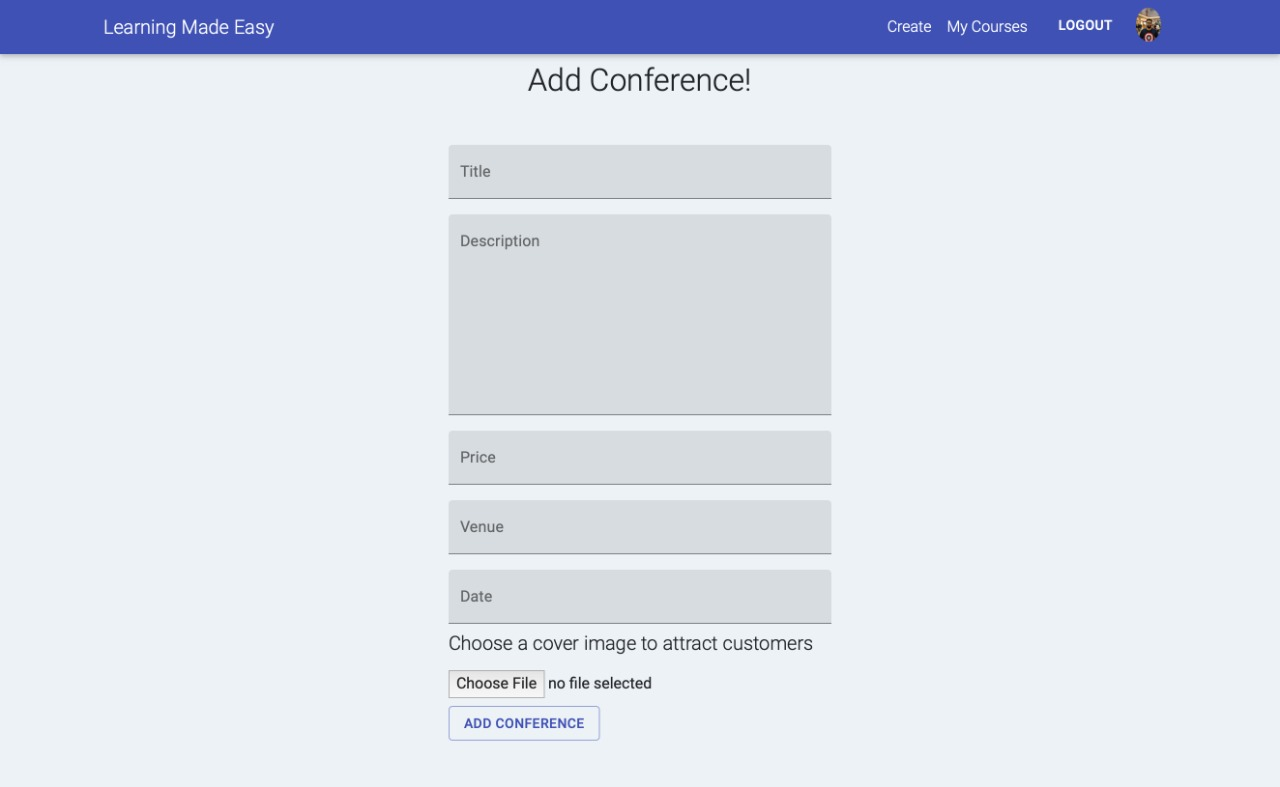
Course Details:



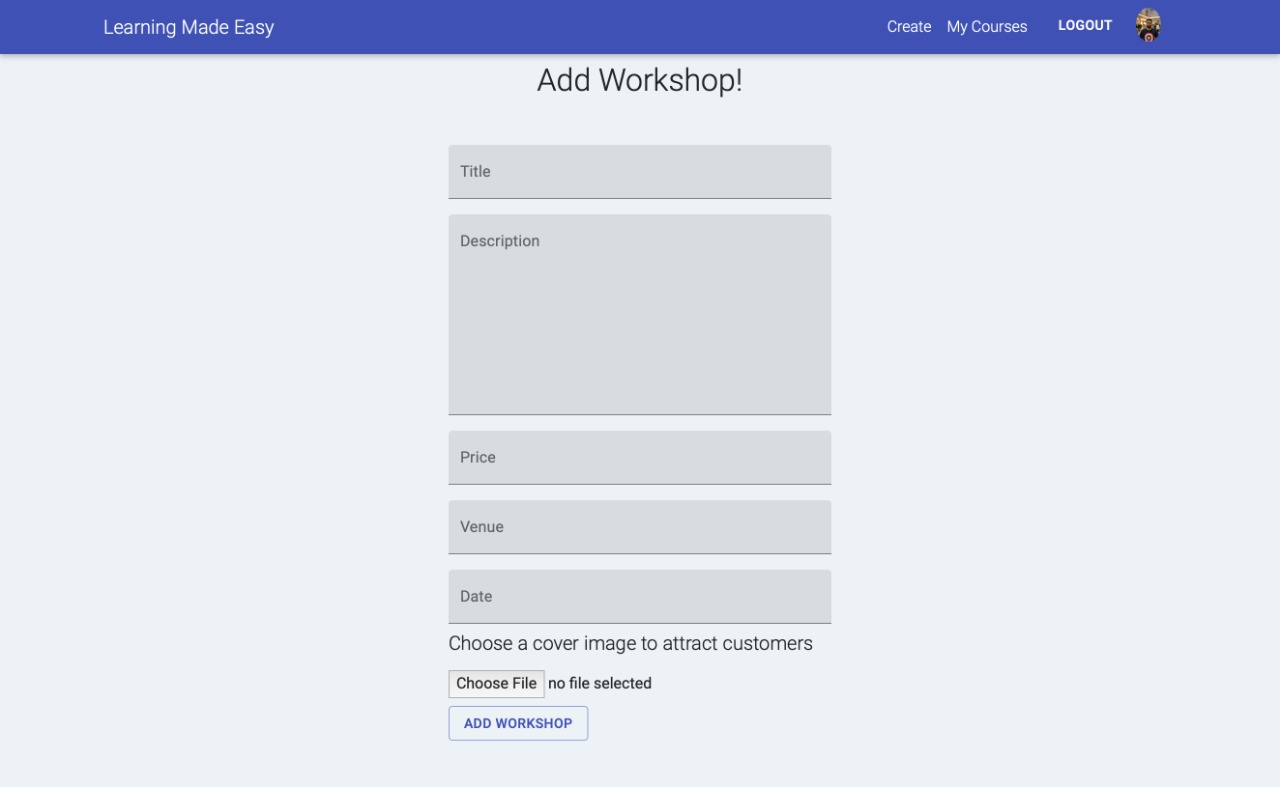
Admin Panel:



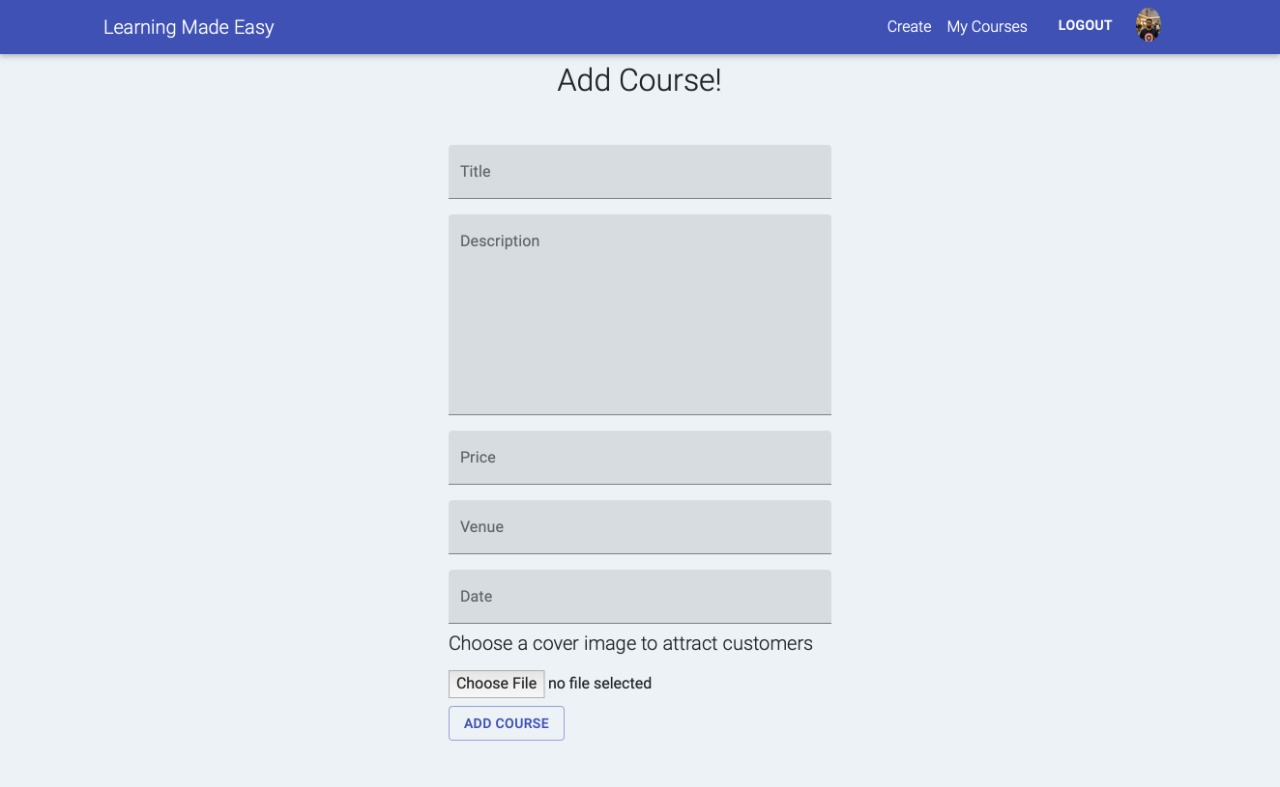
Adding a conference (Admin):



Adding a workshop (Admin):

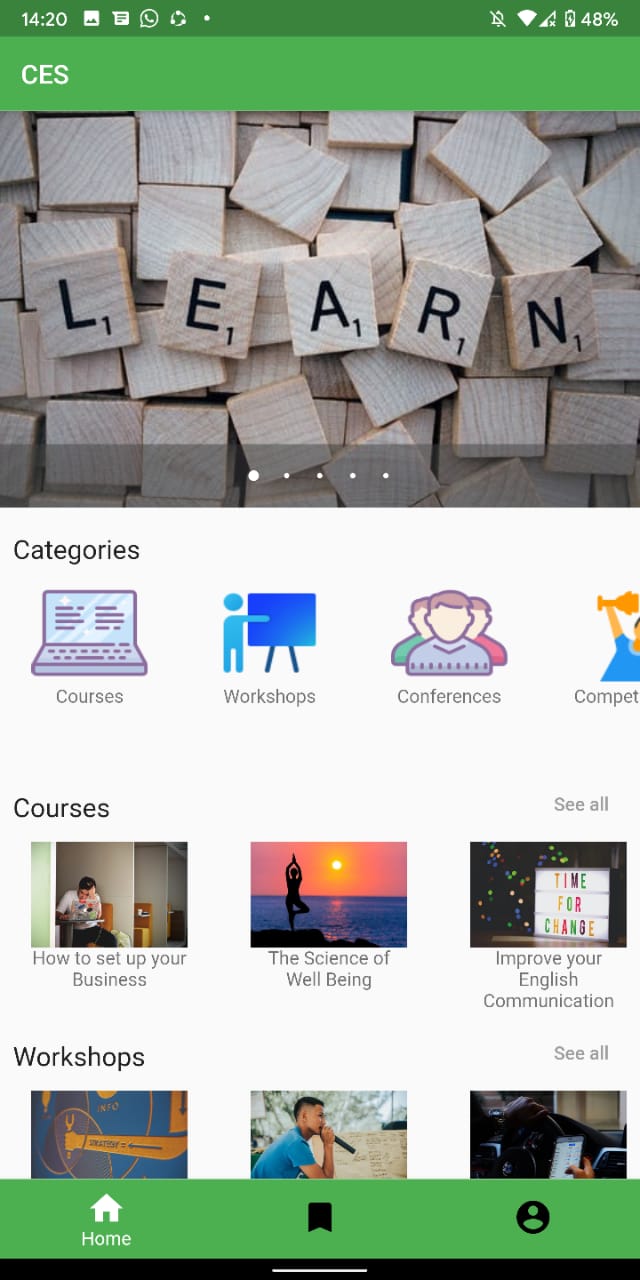


Adding a course (Admin):

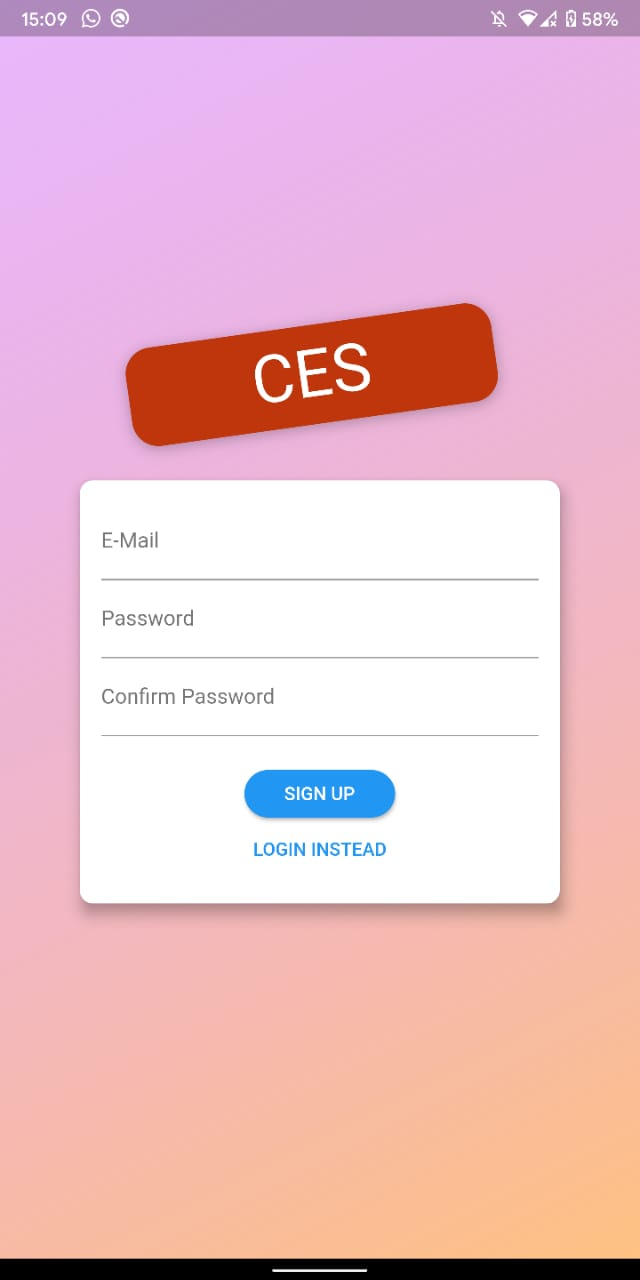


**Mobile App:**

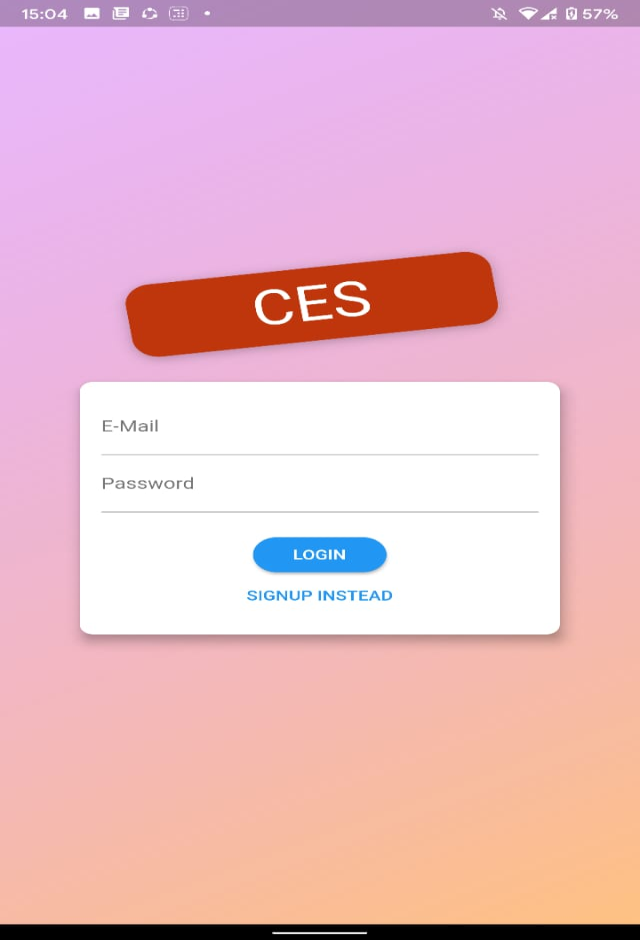
Home Page/Dashboard:



Registration:



Sign in:



List of courses/workshops/conferences:



Course Details:



**3.2 Hardware interfaces**

* Mobile devices, laptops and home desktops are supported (Any small, medium and large screen devices that have internet access and web browsers).
* The data and control interactions for the flow of data between students and admin will be handled by caretaker of the particular courses/workshops/conferences.
* The data obtained from the users are stored on the server side where database is created with appropriate tables.

**3.3 Software interfaces**

* Database: MongoDB
* Frontend: ReactJS
* Operating System: developed on Linus platform (Ubuntu), executes on Windows (XP and above).

**3.4 Communication interfaces**

* Communication standard : Hyper Text Transfer Protocol (HTTP)
* Security Standards: Website is not secured. HTTPS certificate and other secured encryptions yet to be assigned.
* Web Browser: HTML5 Compliant web browser

1. **CONCLUSION**

This website/mobile application designed using MERN stack and Flutter does all the interaction between students and admin’s of the particular course/workshop/conferences respectively. It has two views i.e. student’s user profile and the admin’s view. Our website/mobile application gains the organization to interact and get connected with many number of students and educate them, guide them and make their careers.