



# Feasibility Study Document

**Project Title:** University Course Automation

**Subject:** Software Engineering

**Course Code:** UE15CS402

**Team Name:** SE-B6

**Section:** B

## Team Members

Name	USN
Bharatraj Telkar	01FB15ECS066
Bhavana Naga Sai Malepaty	01FB15ECS068
Chitra Moger	01FB15ECS081
Danish Sultan	01FB15ECS087
Devaraj V Nayak	01FB15ECS092
Durga Akhil Mundroy	01FB15ECS097
Gowtham J A	01FB15ECS109
Gururaj Kulkarni	01FB15ECS113
Hardik Mahipal Surana (Coordinator)	01FB15ECS116
Vishwanath P	01FB16ECS727

# Table of Contents

<b>Table of Contents</b>	<b>2</b>
<b>Problem Statement</b>	<b>3</b>
<b>Executive Summary</b>	<b>3</b>
<b>Current Systems and Processes</b>	<b>3</b>
Current Operations	3
Physical Environment	3
User Organization	4
<b>System Objectives</b>	<b>4</b>
Description of Products and Services	4
High level Block diagram showing the solution	5
Targeted Customers and Benefits	5
Technology Considerations	5
Product/Service Marketplace	5
Marketing Strategy	6
Schedule	6
Issues	6
Technical	6
Implementation	7
Assumptions and Constraints	7
Assumptions	7
Constraints	7
<b>Alternatives</b>	<b>7</b>
Maintain the Status Quo	7
Alternative - Use Existing Infrastructure	7
<b>Findings and Recommendations</b>	<b>8</b>
<b>Project Objectives</b>	<b>8</b>
<b>Project Plan</b>	<b>9</b>

# Problem Statement

The current course conduction and management ecosystem is highly manual, mismanaged and cumbersome for both teachers and students. There is no easy-to-use, configurable, centralized mechanism or platform which can be leveraged for assistance. Hence, this project proposes to provide a cohesive experience for course management including features like constructing a course, accommodating students, sharing content, evaluating assignments and so on.

## Executive Summary

This project aims to provide a centralized solution for creation and management of courses at higher educational institutions in India. The solution is provided as a web application and is deployed based on a subscription or licensing model. It will initially be deployed in PES University in Bangalore, India. The users of this product are students, teachers and university administrators, each of whom will have access to specific functionalities and will have well-defined privileges.

## Current Systems and Processes

### Current Operations

The current process of overall management of a course is highly manual, decentralized and does not follow any form of standardization. This develops a rigid format which once prepared is not conducive to adaptation by teachers. An example for this is creation of course content. Once the topics to be taught is prepared, teachers cannot change it even though over the course's timeline the teacher finds it necessary to do so. There is also no system in place for storage and dispersal of important course-related and generic information.

### Physical Environment

For updating students about the course information, asking them to choose electives etc. the existing system heavily relies on the usage of email and online forms which are not reliable and are difficult to manage and store. For enrolling students in a course, online forms are used which becomes difficult to manage at scale. Also, data such as attendance is infrequently updated on an existing online platform which makes it difficult for students to have the latest information. Submission of assignments is also highly manual and is not scalable.

## User Organization

There are three categories of users in the current system:

1. Students: They study the offered courses, attend classes and submit assignments.
2. Teachers: They teach/instruct the courses and are responsible for its conduction and management which includes creating and sharing the content, managing students and their doubts and evaluating their assignments.
3. Administrators: They orchestrate the seamless functioning of all courses, students and teachers.

## System Objectives

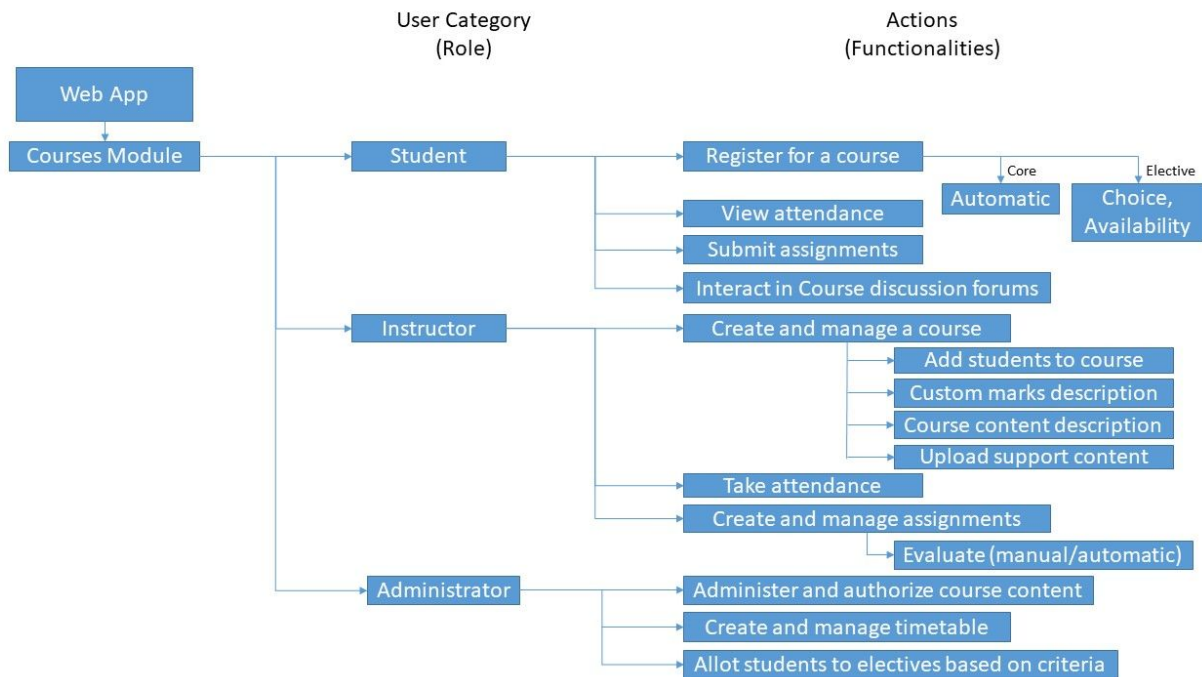
The proposed system is expected to automate the overall management of a course and establish a centralized storage solution for all course related information. The solution will be exposed through a web interface to all user categories. The system is expected to be user-friendly, intuitive, engaging, informative and reliable.

The objectives and high-level requirements the system is designed to meet along with supported functionalities can be found in the block diagram mentioned below. The system will be fully automated for managing student and course related data. It will also include some manual activities such as uploading of assignments, awarding marks to students and verifying student enrolment in courses.

## Description of Products and Services

The Courses module will provide a centralized, user-friendly alternative to the traditional way of course management. The allocation of courses, which usually takes a few weeks to finalize, can be done in a relatively shorter duration of time. Further, because of the integration of attendance with the course, it will provide a smarter, dynamic alternative to the practice of keeping specific registers for each section. The module also provides a uniform way to conduct and evaluate assignments, share resources used in the class and also discuss the subject in dedicated discussion forums.

## High level Block Diagram



## Targeted Customers and Benefits

The entire spectrum of users, namely students, teachers and administrators will benefit from this technology. Students will be provided a centralized hub for all things related to courses. Efficient course creation and management will save hundreds of manual work hours for teachers and administrators which improves overall productivity.

## Technology Considerations

We will be using the MERN stack which stands for MongoDB, ExpressJS, ReactJS and NodeJS. These are all languages used to design, build, and implement interactive web applications. These languages enable us to create authentication systems, interactive information collection and dispersion systems and productionize efficient interfaces across devices. Alongside this, we shall deploy our web application using either hardware provided by our sponsor or by using existing cloud deployment solutions.

## Product/Service Marketplace

The project aims to provide a social learning platform for all stakeholders. The existing annual earning potential for similar solutions in the market is more than \$10 million (estimates gathered from a single company's earnings report). The target market consists of around 36,000 higher education institutions with a total stakeholder population of 25 million people in India alone.

Existing market players who stand as our competitors include companies like Edmodo, Instructure, Moodle, Snapwiz etc. most of which are based out of USA.

## Marketing Strategy

The product, which is the web application provided as a service will be distributed to education institutions via a subscription or license based model. Our customers long for simple, hassle-free and secure management of educational content and confidential information, which is exactly what we deliver. This is also the key differentiator for our competition. Based on the satisfaction of our existing customers, word of mouth marketing will prove to be an important tool for our future organic growth. We also strive to have a healthy feedback loop with our customers which helps us to improve continuously. Moreover, being students, our unique perspective provides us with valuable, on-ground intel regarding the personalized needs of our customers.

Active endorsements with educational events and institutions is another tool we will use. Through these engagements, we will provide our services for free as a part of a trial scheme. On using the platform and realizing its effectiveness in solving their problems, the users will convert to be our customers.

## Schedule

Deadline	Task
September 22nd 2018	Complete requirements and design documents
October 6th 2018	Sprint 1 Features + Testing
October 27th 2018	Sprint 2 Features + Testing
November 10th 2018	Sprint 3 Features + Testing
November 17th 2018	Final Submission

The schedule is subject to change depending on the issues included in the sprints and we aim to have shorter and more sprints.

## Issues

### Technical

- Since most of the team is not familiar with the MERN stack, it might result in slow initial progress.
- Integrating new features with existing user interfaces

## Implementation

- Users with low Internet bandwidth may have difficulty in seamlessly utilizing the portal
- Unauthorized personnel may get unsolicited access to data
- Handling high amounts of website traffic when many users try to access the portal
- Servers can be affected by virus

## Assumptions and Constraints

### Assumptions

- The authentication module already prepared by the web team in ALCODING club has implemented the required features to satisfactory levels.
- Due to an already existing database design, any modifications or additions we make must not break the rest of the system.
- Initial deployment of the app will be limited due to lack of hardware resources in the club.

### Constraints

- The project timeline is 4 months.
- All features need to be well-tested and must be scalable.
- Individuals utilizing the web app must possess a system of Windows 7 or greater to support the modern technologies being used as a part of the web stack.

## Alternatives

### Maintain the Status Quo

Maintain the status-quo, i.e., we continue to use the proven, time-tested method where everything is done manually. We realize that the chances for human error are maximum. The process takes a number of weeks just to allocate courses to students. The tasks which form an integral part of the course framework like conducting tests and assignments, grading students, sharing resources and marking attendance are completely isolated and performed independently, leading to delayed and incomplete display of information related to the same.

### Alternative - Use Existing Infrastructure

We can create a module only to automate the course allotment process. All the other course-related tasks can be circulated using available applications.

For example:

- PESU Academy application can be used to share the following: attendance, assessment marks, courses allotted and the time-table.
- Impartus is a third-party application where lecture videos are captured and shared.

- Some instructors share resources on google drive while others email the resources to the students using google groups.
- Students mail their assignments to the relevant faculty.

This alternative solution solves the cumbersome process being followed for elective registration in our institution, but other course-related tasks will be spread out and still be manual, with the PESU application only displaying the numbers fed into it.

## Findings and Recommendations

Based on the information presented in this feasibility study, we come to the conclusion that the present procedure being followed with respect to the allocation and conduction of courses involves a lot of unnecessary manual work, is highly decentralized, unorganized and time consuming. Thus, it is recommended to utilize the technology presently available to alleviate the shortcomings as described earlier. Our module will be the go-to destination for everything related to courses.

## Project Objectives

The main objective of this project is automate the administration work involved in conducting a course.

- Development and Implementation Issues:
  - Since the ALCODING club has already developed a different part of the web app, we must make sure to have clean interfaces throughout the app.
  - It will be challenging to artificially test the scalability of the system in high load situations.
  - Since most of the team is not familiar with the MERN stack, it might result in slow initial progress.
- Issues concerning Assumptions, Constraints, and Limitations:
  - We assume that authentication scheme is already implemented by the ALCODING club.
  - Due to an already existing database design, any modifications or additions we make must not break the rest of the system.
  - Initial deployment of the app will be limited due to lack of hardware resources in the club.
- Hardware and Software Alternatives:
  - End product will likely use IaaS to host all services. The application can hence be developed with an architecture that leverages easy scalability. AWS, Google Cloud and Heroku are IaaS alternatives for this.
  - Our choices for software are constrained by the fact that some part of the web app has already been developed. Alternatives like the Python-Django framework or the LAMP stack are feasible too, but there is risk involved in developing an interface.
- Technology:
  - The application will be developed using the MERN stack which brings in a unique learning experience.



- This technology stack is supported by many IaaS providers natively. This not only makes deployment simple, but also helps with easy scalability, diverse IaaS choices and lower risk.
- Marketing:
  - Once fully developed, this app can become the standard way of conducting courses in our institution.
  - Based on inputs from other departments and subsequent improvements, other institutions will realize the incentive of upgrading their old systems to our intuitive web application.
  - The level of automation in this market space is very limited and our app helps fill that gap with a robust course management system.
- Organizational:
  - Since the app can be provided as a SaaS, there is no additional changes to the personnel that is required.
  - With a significant number of users, adequate documentation, tutorials and training material would be required to transition an institution to our app.
- Financial:
  - A lot of investment is required to develop the initial product and to make alterations to it based on user feedback. Once in a stable state, opportunities to expand to other institutions will arise.
  - Although this has significant risks associated with it, once the app is popular in a few institutions, sales will start picking up.

## Project Plan

1. Deliverables of the Project:
  - a. A web app with complete UI and backend based on the negotiated requirements.
  - b. Sufficient documentation to allow easy deployment and scaling of the application.
2. Process Model:
  - a. Due to the modular nature of the different aspects of the project, SCRUM has been chosen as the process model.
  - b. Also, we shall use a shorter sprint period of 1 week instead of the standard 3 weeks. This will help in managing the project and staying accountable for the work being done.
3. Identification of the upstream-downstream partners needed for the product:
  - a. We will have a continuous dialogue with the ALCODING club's web app team regarding various aspects of the development process and requirements.
4. Resources needed for the project/product
  - a. During development, we will require a git repository for source code management. We will likely use GitHub and hence tailor our development process around it.
  - b. For deployment, we will either require physical servers or IaaS providers.
5. How are you organizing your team in the project?
  - a. The team will have a flat hierarchy.
  - b. Most of the requirements will be broken down into small GitHub issues that one person will be able to take ownership of. This involves seeing that issue through till

the end by discussing various techniques, fixing the issue, writing tests and also by being involved in the review process.

6. Standards-Guidelines-Procedures:

- a. For coding standards, we aim to follow the industry leaders. Most of them have their style guides openly available. To enforce these standard, code linters shall be setup as part of the testing procedures.
- b. These standards shall also include instructions to be followed during code management, like the commit style guide.
- c. Guidelines to contribute to the project shall be well documented and all team members should follow those guidelines while working on an issue.

7. Communication Mechanism:

- a. Whatsapp: For informal discussion involving only the team members
- b. EMail/Google Group: For formal messages to Sponsor other external partners.

8. Quality Criteria:

- a. Quality will be assured by having strict testing procedures in place during development.
- b. Also, frequent meetings with ALCODING club will ensure that we never stray from the agreed upon quality of the deliverables.

9. Work Packages:

- a. The list of requirements along with the design document will be converted to a list of issues which can be assigned to team members.
- b. Also, for each issue being solved, the team has to take up peer reviews to make sure that the code being pushed has the requisite quality.

10. Delivery Means:

- a. Project Source code and documentation will be delivered through the open source GitHub repository.