**Systems studied: Moodle and StudioHQ**

**Target Users:**

1. Students
2. Lecturer (Admin)
3. Tutors

**Lecturer + Tutor**: Combined (Web application)

**Students**: Web + Mobile Application

**Backend**: Hosts all the content online and provides RESTful APIs

**Documentation**: For users and developers

**For Students**:

* File upload and download
* Notification system
* Dashboard to view all courses and important announcements
* Ability to bookmark external resources to a certain course/file
* Chat capability with online users
* Project groupings and document sharing
* Private files within a project
* Calendar view with color differentiation
* See personal track record over the semester
* Social forum to discuss ideas and clarify doubts
* Ability to perform in-class interactive activities such as drawing (add-on)
* Easy reference to slides, pages for emailing (add-on)
* Create tests and challenges for peers (add-on)

**For Lecturer**:

* File upload and download
* Administrator privileges
* Create student accounts or generate common password and let students register on their own
* Staffroom to post announcements or discuss course material with co-lecturers or tutors
* Dashboard to view all the courses being taught
* Ability to send notifications
* Ability to CRUD all content and specify levels of access to different user groups
* Chat capability with online users
* Add/delete/edit grades
* Ability to create interactive activities
* Ability to record/screencast lectures and write real-time comments (add-on)
* Create online timed tests/quizzes, verify the answers and automatically update the grade book (add-on)
* Easy reference to slides, pages for emailing (add-on)

\* Tutors are a subset of lecturers

**For Backend**:

* Provide RESTful APIs to execute above mentioned functions
* Store the dynamically/statically generated content in the database efficiently
* Efficient recovery of data
* Regular backups and server-side analytics to understand the problems (if any)
* Extremely robust to handle high amounts of traffic and display data in real-time.
* Encryption of sensitive data

**For Documentation**:

* Everyone needs to document their work in such a way that it is easy to understand and make changes to the system without much hassle.
* Web and mobile application documentation must be done two ways, for users and for developers.
* Backend documentation is only for developers.
* Demo videos for the most important functions of the system for better user understanding

**Code and document sharing platform**: GitHub

**Task planning and scheduling**: Teamweek