



# CLASS CONNECT

JAYME EATON

ROHAN KADKOL

ANNABELLE KANCHIRATHINGAL

MICHAEL MOREL

CHAMYRIA MUSE

HARON TEMAM

# Product Vision Card

**FOR** students

**WHO** require a powerful learning management tool

The **ClassConnect** application is an educational software

**THAT** provides a way for students to access their course material and organize their complete school life.

**UNLIKE** Moodle that offers only minimal and small-scale course features,

**OUR PRODUCT** offers powerful course features, large-scale campus activity features, to provide one of the most comprehensive tools for students.

**FOR** instructors

**WHO** require a powerful course administration tool

The **ClassConnect** application is an educational software

**THAT** empowers instructors with tools to share course documents, conduct and grade exams, and communicate with their class.

**UNLIKE** Moodle that provides a clunky instructor-side UI, slow website load-times, anti-intuitive website hierarchy, and tons of outdated impertinent features,

**OUR PRODUCT** provides a rich, immersive UI experience with blazingly fast load times, ergonomic website hierarchy, and tons of essential modern features that all come together to effectively raise the bar for the ideal classroom experience

## User Stories:

**User Story #1:** As a student, I want an easy way to view my upcoming assignments and their deadlines, so that I can prioritize which assignments to work on according to their time remaining and predicted time required.

**User Story #2:** As a student, I want a digital calendar which updates with assignment deadlines, so that I can stay on top of my work

**User Story #3:** As a student, I want a public Q&A section for both students & instructors, so that I can see the instructors' answers to general questions and get help from other students as well.

**User Story #4:** As a student, I want to be able to personalize notifications based on my set study schedule to finish my work before the deadlines, so that I don't waste time or forget a deadline.

**User Story #5:** As a student, I want to find all important information like midterm dates, grading scales, instructor's email, etc. for all my classes extremely easily, so that I do not have to go search for it in the syllabus or any other documents that the instructor uploads.

**User Story #6:** As a student, I want all my online classes to have an in-built Zoom client with Zoom links and past recordings in ClassConnect itself, so I can save time searching in multiple places (Moodle or its equivalent, Syllabus, Outlook) before each class.

**User Story #7:** As an instructor, I want to plan my entire course in detail using a tech tool, so that I can provide students with an interactable and personalized class schedule instead of a static schedule table in the syllabus PDF.

**User Story #8:** As a student, I want a faster way to send messages to instructors than using email, so that myself and instructors do not have to spend as much time on messaging.

**User Story #9:** As an instructor, I want a button in the chat UI to go directly to the next message, so that I may respond to my students as fast as possible.

**User Story #10:** As an instructor, I want my incoming chat messages organized with oldest and marked urgent messages at the top with non-urgent recent messages at the bottom, so that I can respond to the most important messages first.

**User Story #11:** As a student, I want to mark my messages as urgent when needed, so that my instructor can respond to my important messages before other non-urgent ones.

**User Story #12:** As an instructor, I want to send short pre-made messages to answer students' chats when needed, so that I can respond and move to the next message with one action.

## Functional Requirements Specification:

Item	R-1: Course facilitators can upload documents
Summary	Course facilitators upload slides, homework questions and other documents as needed
Rational	Course facilitators should be able to share important documents with students in an effortless manner. Students should be able to access them without much difficulty.
Users	Course Owner, Course facilitators (Instructors, TAs, graders). One of the facilitators must be the owner
Pre-conditions	<ul style="list-style-type: none"> <li>• User is logged in</li> </ul>
Basic Course of Events	<ul style="list-style-type: none"> <li>• User navigates to the course page</li> <li>• User clicks the "add" button and selects the document type from the dropdown (slides, homework, exam). This tag with each document will help students filter their search results</li> <li>• System allows upload from local storage, OneDrive storage, Google Drive storage, and DropBox.</li> <li>• User selects options: <ul style="list-style-type: none"> <li>○ Out time: now (default), schedule for later</li> <li>○ Visibility (multiselect): everyone (default), by student group (e.g., Two sets of exams for each half of the class), facilitators, individually select</li> </ul> </li> <li>• User clicks "Add"</li> </ul>
References	NF-2

Item	R-2: Course facilitators add questions to course question bank
Summary	Course facilitators add all their testables' (exam, homework, and quiz) questions to the course question bank
Rational	Adding questions to a question bank helps course facilitators to quickly construct a testable at any time and reuse questions from old testables
Users	Course owner, Course facilitators (Instructors, TAs, graders)
Pre-conditions	<ul style="list-style-type: none"> <li>• Course facilitator is logged in</li> <li>• Course facilitator has privilege from course owner to modify the course question bank</li> </ul>
Basic Course of Events	<ul style="list-style-type: none"> <li>• User navigates to the question bank page for that course</li> <li>• User clicks the "add" button</li> <li>• System displays a text editor along with options to upload images, videos or files</li> <li>• User writes the question in the text editor in the markdown format using the simplified toolbar that automatically types markdown constructs. <ul style="list-style-type: none"> <li>○ E.g., Adds *...* around the selected text if the user clicks the italics button</li> </ul> </li> <li>• User chooses answer type: short text, long text, multiple choice (single correct), multiple choice (multiple correct) <ul style="list-style-type: none"> <li>○ If the user selects multiple choice, the system allows the user to enter the options</li> </ul> </li> <li>• User adds auto grader conditions (optional): <ul style="list-style-type: none"> <li>○ Short text, long text: List of &lt;regex match condition, score&gt; grading scheme</li> <li>○ Multiple choice (single and double): List of &lt;option, score&gt; grading scheme</li> </ul> </li> <li>• User clicks "Add"</li> </ul>
References	R-7, R-8, R-9, NF-2

Item	R-3: Students ask questions in a Q&A forum
Summary	Students can ask questions in a dedicated Q&A forum to get help from course facilitators and other students
Rational	Class emails get lost in a sea of personal emails, promotional emails, and calendar invites. This feature introduces a dedicated Q&A forum for each course section to keep course questions organized and prevent clutter. It will also allow students to answer questions and help other students.
Users	Course owner, Course facilitators (to ask themselves questions to then share the answer with other students having the same question), students in that course
Pre-conditions	<ul style="list-style-type: none"> <li>• User is logged in</li> <li>• User is enrolled in course</li> </ul>
Basic Course of Events	<ul style="list-style-type: none"> <li>• User clicks the "Questions" button in their course page</li> <li>• System presents user with publicly shared FAQs and answers</li> <li>• If the user still has a question, then can click the "Ask a question" button</li> <li>• System presents user with a simplified markdown editor and options to attach pictures, videos, or other files</li> <li>• User enters question with relevant attached resources</li> <li>• User selects options: <ul style="list-style-type: none"> <li>○ Relevant subject: Tagging the topics, homework's, exams, or slides helps other students filter questions when seeing if it is already answered.</li> <li>○ Relevant topic: Tagging the course topic</li> </ul> </li> <li>• User clicks the "Submit" button</li> </ul>
References	NF-3

Item	R-4: User can post activities
Summary	Users post the activity they are planning to do in order to find more people that match their interest in that activity. E.g. A user can post that they are playing frisbee in the parade grounds. People with similar interests will get notified and can thus join.
Rational	Student clubs host well-organized large-scale activities. People want regular small-scale events for small fun activities that are accommodating and require no planning. E.g., Playing cornhole on the parade grounds. This feature will bridge that gap.
Users	Users in that organization
Pre-conditions	<ul style="list-style-type: none"> <li>• User is logged in</li> <li>• User belongs to that organization</li> </ul>
Basic Course of Events	<ul style="list-style-type: none"> <li>• User clicks the "Social" icon/button</li> <li>• System displays UI with the add activity form</li> <li>• User selects options: <ul style="list-style-type: none"> <li>○ Category (single-select): Sport, Watch party, Educational</li> <li>○ Tags (multi-select): Large dropdown with tags like soccer, frisbee, French, history, and video game. Others who subscribe to these tags will be notified about this event</li> <li>○ Date/Time</li> <li>○ Location</li> <li>○ What to bring?</li> </ul> </li> <li>• User clicks the "Submit" button</li> </ul>
References	R-5, NF-2, NF-3



Item	R-5: User subscribes to activity tags
Summary	People can find other people doing activities they like by subscribing to tags that match their interest. They can then be notified when someone is doing an activity.
Rational	Finding people that match our interests is always difficult. This is true in large universities and schools. This FR helps students find others in the same organization faster than any other application.
Users	Users in that organization
Pre-conditions	<ul style="list-style-type: none"> <li>• User is logged in</li> <li>• User belongs to that organization</li> </ul>
Basic Course of Events	<ul style="list-style-type: none"> <li>• User clicks the "Social" icon/button</li> <li>• System displays dashboard with "Events you may like", "Trending right now", "What your friends are attending"</li> <li>• User clicks the "settings" icon/button</li> <li>• System shows the settings list</li> <li>• User can add interested activity tags in the preferences section</li> <li>• User clicks "save changes"</li> </ul>
References	R-4, NF-3

Item	R-6: Students and course facilitators communicate with private messages
Summary	Students and course facilitators can start private message threads.
Rational	Emails are a great way for student-course facilitator private communication. However, they are slow to get replies and untidy when the message thread becomes large. This FR brings a feature that is as quick as texting or Discord, but with more education-related messaging features
Users	Course owner, Course facilitators, students in that course
Pre-conditions	<ul style="list-style-type: none"> <li>• User is logged in</li> <li>• User belongs to that course</li> </ul>
Basic Course of Events	<ul style="list-style-type: none"> <li>• User clicks the "Messaging" button on the course page</li> <li>• System displays the list of previous messages</li> <li>• User clicks the "Compose button"</li> <li>• System displays a composed message form</li> <li>• User chooses options: <ul style="list-style-type: none"> <li>○ To (multi-select): Any/Multiple course facilitators</li> <li>○ Importance: regular priority (default), urgent</li> </ul> </li> <li>• User enters subject and the content of the message in markdown format</li> <li>• User attaches relevant pictures, videos, or other documents</li> <li>• User clicks the "send button" to send the message</li> </ul>
References	NF-2, NF-3

Item	R-7: Course facilitators can create testables (exams, quizzes, homework assignments)
Summary	Course facilitators can choose questions from the question bank and quickly make a testable. They can then add it to the class testables page or export it as a PDF for a paper version testable.
Rational	Course facilitators should be able to give testables to students to test the students' understanding of the material
Users	Course owner, course facilitators
Pre-conditions	<ul style="list-style-type: none"> <li>• User must be logged in</li> </ul>
Basic Course of Events	<ul style="list-style-type: none"> <li>• User clicks the testables section on the course page</li> <li>• System shows a list of previous testables</li> <li>• User clicks "Add"</li> <li>• Systems displays an "add testables" form</li> <li>• User chooses options: <ul style="list-style-type: none"> <li>○ Category: exam, quiz, homework</li> <li>○ Location: online during-class (default), online, paper-and-pen</li> <li>○ Duration: unlimited, custom time</li> <li>○ Release: now (default), schedule for later</li> <li>○ Out: now (default), custom time</li> <li>○ Deadline: custom time</li> <li>○ Attempts: 1 (default), custom number</li> </ul> </li> <li>• User clicks the "add questions" button</li> <li>• Systems shows a list of questions from the question bank along with an "add new question" button</li> <li>• User can select multiple questions</li> <li>• (Clicking "add new question" will take them back to the question bank. This case is handled in R-2)</li> <li>• User clicks the "add testable" button to add the testable</li> </ul>
References	R-2, R-8, R-9, NF-3

Item	R-8: Students work on testables
Summary	Students work on the testables assigned to them by the course facilitators
Rational	Students should be able to work on the testables that the course facilitators assign to them
Users	Users enrolled in that course
Pre-conditions	<ul style="list-style-type: none"> <li>• User is logged in</li> <li>• User is enrolled in course</li> </ul>
Basic Course of Events	<ul style="list-style-type: none"> <li>• Student clicks on the testables section on the course page</li> <li>• System displays the testables that are "out"</li> <li>• Student clicks on the appropriate testable</li> <li>• Systems displays the options that the course facilitator set for the testable along with a "start attempt" button</li> <li>• Student clicks the "start attempt button"</li> <li>• System displays all the questions in the testable along with the timer with the time left to finish the attempt</li> <li>• Students can answer questions by selecting options, writing in the textbox, or attaching images and documents, depending on the type of answer expected.</li> <li>• When done, students can click the "Submit" button to submit their work</li> <li>• When the timer runs out, the current answers will automatically be submitted</li> </ul>
References	R-2, R-7, R-9, NF-3

Item	R-9: Course facilitator grades testables' attempts
Summary	Course facilitator assigns grade to students' testables
Rational	Course facilitator should be able to grade students' testables and release the results to the student
Users	Course owner, Course Facilitators
Pre-conditions	<ul style="list-style-type: none"> <li>• Course facilitator is logged in</li> </ul>
Basic Course of Events	<ul style="list-style-type: none"> <li>• User clicks on the testables section on the course page</li> <li>• System displays all the testables</li> <li>• User clicks on the testable that they want to grade</li> <li>• System displays the submission statistics (number of submissions, number of late submissions, average time taken per attempt, average number of attempts) and the options to grade by: <ul style="list-style-type: none"> <li>○ By student: Grade student's work one at a time</li> <li>○ By question: Grade each question from all students one at a time</li> </ul> </li> <li>• User chooses an option to grade</li> <li>• System displays each student's entire attempt (if selected "by student") or all attempts for the selected question (if selected "by question")</li> <li>• Users can click on the apply auto grader conditions for each question/attempt to grade the answers using the auto-grading conditions attached with the question (As discussed in R-2)</li> <li>• User can also manually modify the auto grader's assigned grade for each question</li> <li>• User clicks the "next" button to go to the next question/attempt</li> </ul>
References	R-2, R-7, R-8, NF-3

Item	R-10: Students see personal calendar
Summary	Students can see a personal calendar with all their classes, deadlines, testables, and activities. This calendar should be displayed in two modes: Regular and Gantt
Rational	Dates and times are quickly forgotten. Students need a calendar tool that is built into their education platform itself that automatically syncs with their school life events to that they don't miss any important deadline.
Users	Students in that organization
Pre-conditions	<ul style="list-style-type: none"> <li>• Student is logged in</li> </ul>
Basic Course of Events	<ul style="list-style-type: none"> <li>• Student clicks the "My calendar" button</li> <li>• System displays calendar along with a toggle switch for "Regular" and "Gantt" <ul style="list-style-type: none"> <li>○ Regular: shows each event's end date on the calendar</li> <li>○ Gantt: shows each event's start and end date in a swim lane</li> </ul> </li> </ul>
References	NF-2, NF-3, NF-4

## Non-Functional Requirements:

Item	NF-1: Time constraint for system maintenance
Summary	Mean time to change for defects should take less than 5 business days
Rational	The shorter the MTTC can be for the application, the faster the application can be put back into the users' hands. This also allows ClassConnect the ability to quickly react to market changes.
Requirements	<p>Redundant system with second server to ensure all users still have access to main features of application main server is being updated</p> <ul style="list-style-type: none"> <li>• Reports for users to log any bugs that they find or other issues that may need to be fixed</li> </ul>
References	NF-2

Item	NF-2: System Recovery
Summary	Initial database runs full backups monthly with increment backups bi-weekly.
Rational	The application will hold sensitive information such as grades, user profile information and assignments which can be updated daily. So, it is best for the application to keep a backup of such information in case of a system reboot or security threats.
Requirements	<ul style="list-style-type: none"> <li>• Data must be encrypted with encryption algorithm prior to backup</li> <li>• Disaster recovery test is executed every 6 months to ensure system recovery and backup are working properly.</li> </ul>
References	R-1, R-2, R-4, R-7

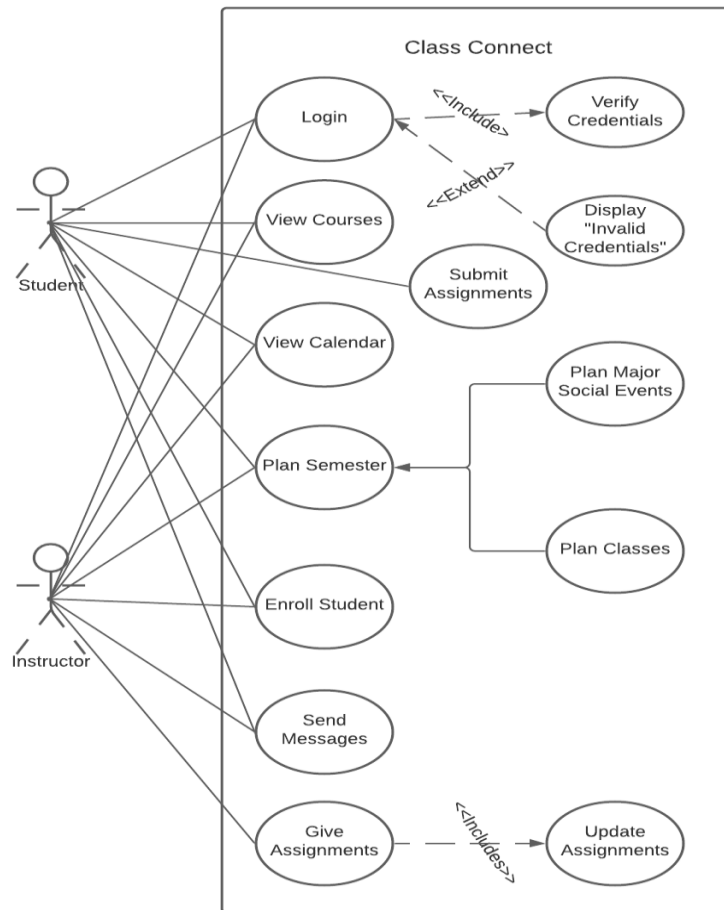
Item	NF-3: Usability of Application
Summary	85% of novice users can learn how to operate main features without outside assistance.
Rational	Making the application learnable by novice users separates us from competitors who have confusing applications and allows users to rely on ClassConnect.
Requirements	<ul style="list-style-type: none"> <li>• Pages must have a uniform format (uniformity in layout and theme)</li> <li>• Each individual page must only consist of one main feature at most</li> </ul>
References	R-3, R-4, R-5, R-6, R-8, R-10

Item	NF-4: Localization: Time Zones
Summary	The UI can switch between Central and Pacific time zones without recompiling or rebuilding the program
Rational	For students that have online classes in different states, it can be hard to keep with deadlines for assignments, this feature would make it possible for students to keep with their assignments in a timely manner.
Requirements	<ul style="list-style-type: none"> <li>• The UI will use cellular connection to detect the time zone the user is in</li> <li>• The user will also be able to manually input their time zone into the application</li> </ul>
References	NF-2, R-10

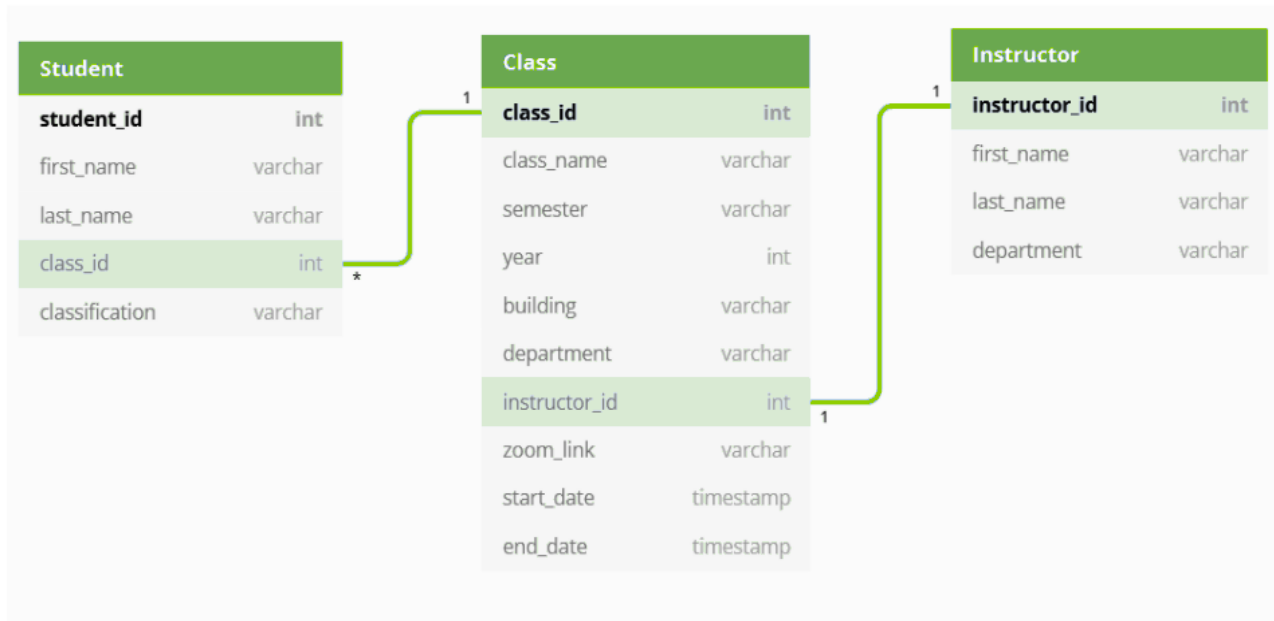


# Models:

## Use Case Diagram



## Database Starter Schema



## Statement of Contribution:

**Jayme Eaton:** Worked on the clarity and formatting of the document and helped with the creation of non-functional requirements.

**Rohan Kadkol:** Worked on the functional requirements. Wrote 4 user stories. Complied and modified all brainstorming user stories in a structured way for this document.

**Annabelle Kanchirathingal:** Designed cover page and helped with the creation of non-functional requirements.

**Michael Morel:** I put together some user stories, helped with format and wording.

**Chamyria Muse:** I worked on the product vision card for the students and professors.

**Haron Temam:** Created the database starter schema and use case diagrams.