# Scope®

## Student's Project
## Performance Management Tool

### Team #10

# Cindy Carrillo

# Abraham Tesfay

# Yulong Ran

San Jose State University     Fall 2019     CS157A

# Project Overview

The main purpose of our project is to develop a web application platform that can be used by schools to provide student's performance review based on sets of project requirements the school or instructor's set.  Our web application platform called "Scope" will ashier a new opportunity for transparency, improvement, and growth among students and instructors.

## The Problem

As technology leaps into the next level, the way education is delivered and measured is also rapidly evolving. The traditional way of grading and testing to measure student skills is slowly fading and upgrading with more concrete and skilled based projects that are meant to make students ready for the 'real world'. In real workplace scenario it's a common practice to provide employees with their quarterly work performance review for growth, improvement, and learning opportunity.

At the moment, almost all educational institutes including San Jose State University don't have a tool to track, measure, or provide feedback on student's project performance. Most team projects done by students are usually tracked and reviewed internally by the professor and this creates a huge gap for transparency, growth and an opportunity for improvement.

## Why 'Scope'

Scope will usher a newer, easier, faster and dynamic way to deliver students and instructors performance review based on different sets of criteria and milestones that the instructors and the students set. Students will have an opportunity to peer review their team, instructor's and reflect on their project performance for better growth and  improvement.

# System Requirement
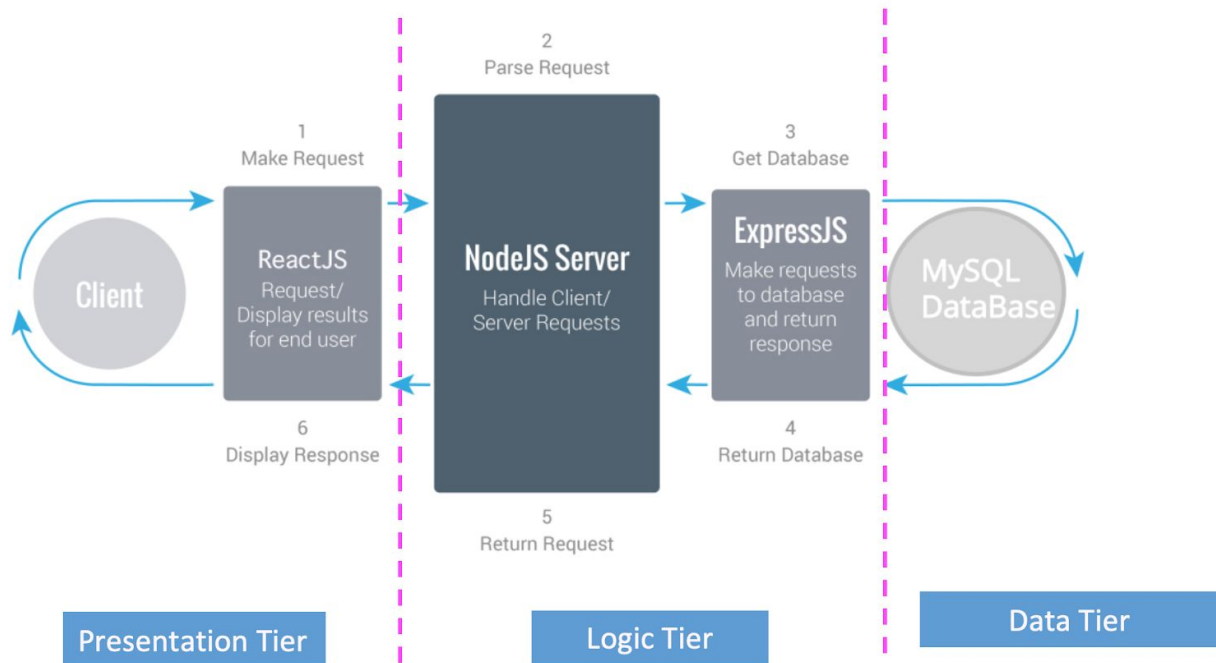


**Presentation Tier:**

In Scope, the user interface will be a web application that handles the interaction with the user and display information. Our team will build this web application with the framework "React" and programming language JavaScript.

**Logic Tier:**

Scope will use Node.JS as a backend server for handling user requests and communicate with the database.

**Data Tier:**

Our project will use MySQL as our RDBMS(Relational Database Manage System) to manage and provide access to the data.

# Functional Requirements

## Users

The users of Scope will be college students who are working on projects for their courses, and instructors who assign the coursework. Both users would then have their performance reviewed and can adjust their academic performance, goals and work habits accordingly. The user groups would have access to the web application through the university. Instructors can provide links where students can create their one time account that follows them through all their courses at the university that are required to complete their degree.

## Functionality and Features

Scope will contain a homepage where the targeted user can search for their university by school name or code. Scope will also have a login/sign up page where the user can specify if they are an instructor or a student and sign in with university credentials (ID and password) or create an account if they are new. For student users, Scope will include self assessments which students can complete themselves for their courses and class projects. They can also complete peer performance reviews for students they have worked with. Additionally, they'll have access to receive feedback from completed performance reviews from previous teammates they have worked with. The performance reviews would have an option of being anonymous for privacy reasons. For instructors, they have access to view their students' performance reviews to better understand the work habits of those that make up the class. Instructors can search within classes for the list of students in the class and their performance. Instructors will have the ability to create performance reviews for their students based off of their projects in their course and rate if expectations were met.

## Inputs/Outputs

The inputs for Scope would include the entered student and instructor data such as name, login information, and reviews. The projected outputs would be the ability to view and collect performance reviews for the users from the database and display on the user interface.

**User : Instructor**

Display course:

- When a user clicks the Dashboard menu on the navigation bar, a dashboard containing a list of courses will appear on the right side of the web application.
- The system shall retrieve course data from the database and display a list of courses under the instructor's profile.

Add course:

- An instructor user shall be able to add a new course by clicking the plus button. After clicking the plus button, the user shall be able to fill in the name, course Id and course period on the pop-up form.
- The system shall update the database with the new course associated with the instructor and update the user interface with new courses.

Edit / Delete course:

- An instructor user shall be able to delete or edit a course by clicking the menu option on the top right of each displayed course.
- The system shall display a form for the user to edit course information and a warning when user trying to delete a course. The system shall update the database based on the user action.

Display Project details:

- When an instructor user clicks the course in the Dashboard, the Dashboard shall refresh its content and display a list of the projects in the course with its name and time period.
- The system shall retrieve the project name and time period in the database for each project and display on the dashboard as an individual component.

Add Project:

- An instructor user shall have the ability to create a project by clicking the plus button on the top of the dashboard. After clicking the add button, the web will have a pop-up form that asks for the name and time period of the project.
- The system shall update the database with a new project under the course.

Add Performance Review:

- An instructor user shall be able to create an evaluation form by clicking the option under the menu for each project component. After clicking the "creating evaluation form"

button, a new form will pop-up, and instructor user can be able to add a list of millstone, requirements, and feedback followed by a set of checkpoints.

Example:

Team members are respectful to each other.  [ Strongly agree, agree, disagree]
Project milestone 2 [On-time, Late]

### Edit / Delete project:

- An instructor user shall be able to delete or edit a project by clicking the menu option on the top right of each displayed project.
- The system shall display a form for the user to edit project information and a warning when user trying to delete a project. The system shall update the database based on the user action.

### Display Team Information:

- When an instructor user clicks a project component in the Dashboard, the Dashboard shall display a list of teams who participated in the project with their team names and number of members.
- The system shall retrieve a list of teams and their basic team information within the course from the database. The user interface shall refresh the dashboard to display updated information.

### Add / Edit Team:

- An instructor user shall be able to delete or edit a team by clicking the menu option on the top right of each displayed team.
- The system shall display a form for the user to edit team information and a warning when user trying to delete a project. The system shall update the database based on the user action.

### Display Team members:

- When an instructor user clicks the team component in the Dashboard, the Dashboard shall refresh its content and display a list of the students in the team with their name and id.
- The system shall retrieve the student name and id in the database for each student and display on the dashboard as an individual component.

Display Student History:

- An instructor user shall be able to click on the Student component and review student's history project.
- The system shall retrieve the student history projects in the database and display on the dashboard as an individual component.

**User : Student**

Display course:

- Similarly as the instructor's view, when the student clicks the Dashboard menu on the navigation bar, a dashboard containing a list of their current courses will appear on the right side of the web application.
- The system shall retrieve course data from the database and display a list of courses under the student's profile.
- Previous terms and the courses taken those semesters will also be available to view.

Display Project details:

- When a student user clicks the course in the Dashboard, the Dashboard shall refresh its content and display a list of the projects in the course with its name, time period, and additional details.
- The system shall retrieve the project name and time period in the database for each project and display on the dashboard as an individual component.

Add Performance Review:

- A student user shall be able to create an evaluation form by clicking the option under the menu for each project component. The student user will navigate to the dashboard, click teams, then select which team (if more than one is open for the user) and specify which team member to evaluate. After clicking the "creating evaluation form" button, a new form will pop-up, and the student user can fill out the set of questions the instructor set and add additional feedback for peers.

Example:

Team members are respectful to each other.  [Strongly agree, agree, disagree]

Project milestone 2 [On-time, Late]

Display Team Information:

- When a student user clicks a project component in the Dashboard, the Dashboard shall display a list of teams who participated in the project with their team names and number of members. This allows the student to view other students / teams performance.

Display Team members:
- When a student user clicks the team component in the Dashboard, the Dashboard shall refresh its content and display a list of the students in the team with their name and id.
- The system shall retrieve the team members information in the database and display on the dashboard.

Display Student History:
- Student users shall be able to click on other student profiles if they share the course and review student's project history and their previous peer performance reviews.
- The system shall retrieve the student history projects in the database and display on the dashboard as an individual component.

Self Assessment:
- Student user shall be able to add a self assessment for themselves for each project they are assigned by the instructor. Navigating to the dashboard and choosing the self assessment button, the student user will be prompted open projects they are in and can complete a self assessment for each team project.
- The system shall retrieve the student self assessment in the database and display only for the student user to see and reflect.

# Non Functional Requirements

Non-functional requirements include many aspects such as performance, scalability, accessibility, security, etc. Within the time of CS157A course, our team aims to build a three-tier architecture application that provides student's performance review based on their projects within the range of single University.

In terms of scalability, students' performance review will follow them from course to course and school to school. Student's from different Universities will be able to review each others project performance history.

For accessibility, Scope aims to bring high-performance websites for our users. Users can be able to access our site either on their PC or mobile platform. The navigation system for users will be simple, intuitive, and provide enough information that matches User's needs.

For security, we plan to have two different levels of users: Instructor and Student. The instructor has access to create a project requirement, milestone, and evaluation form. Students will have access to evaluate their team members performance after the project.

For GUI mockup and prototype designing, we will be using Adobe's Photoshop and Sketch design and prototyping tools.

## Usability

- User will be able to understand easily the way the program application function.
- User should be able to create project, assign project objectives and review the student's project performance.
- System administrators will be able to use the database to effectively manage projects, student accounts and user information without any error or downtime.
- The application usability is designed to be easily accessed by instructors and students.

- Instructors and students can view projects, deadlines by type, timeline and perform essential database transaction with basic and admin access privileges.
- Administrators with managerial privileges will be able to perform essential transaction to produce variety of projects and assigned the projects to users such as students.

## Reliability

- System will see minimal downtime apart from maintenance and errors.
- Users will not encounter system failures, exceptions will be caught, and users will be given useful information to avoid any error that might arise.
- User's personal  information will be safe, secure, and encrypted.
- There will be easy steps for troubleshooting on both ends of use (Administrator or regular user side).

## Performance

- The system should be able to support larger number of users without affecting any performance of the system operations.
- The system should be able to support a large number of projects and user database transaction simultaneously.
- The database should be able to show the correct project tasks and other related data that specifically assigned the corresponding instructors and students
- Users shall be able to find relevant projects, students and instructors by searching through via the Scope platform.
- Loading the database to the system should take no more than a few seconds per page given the user's machine required the minimal device requirements.

## Supportability

- The system should allow to fix bugs for any competent.

- The system should allow to add additional functions to adapt changes.
- The system should work on any operating systems.
- The system should work on any size of screen.
- For development the system should work on React JS, Node JS, and MySQL runtime environment implemented on locally or on remote cloud server

## Implementation

- This system will implement the Oracle's MySQL database, allowing the user to perform a variety of database transactions related to the project performance tool.
- This system will be written using React JS, MySQL query and JavaScript.
- This system will be hosted by an online web host servicer and also available in local machine
- This system implements adding, editing, updating and deleting of users (Instructors and students), related projects, and projects objectives and milestones.
- The system implements the functionality of producing reports and invoices to keep track of productivity and performance of projects and the stakeholders who are involved.
- The system will save user's activities, profiles, and their tasks in a timely manner that suit the project requirements
- The system will also keep track of the status of projects, tasks and milestones and notify users appropriately.
- The system will also allow administrators of the system to add new users, manage the account and set different privileges.
- The project will show or produce overall report of the status of the projects, timelines and milestones to show overall performance of a class project.

## Interface

- This system will interface via online web application using a remote hosting service or can be implemented using local machine running Node Js web server .
- All users' requests will be handled in a web browser.

## Legal

- This project will not be legally licensed under any professional contracts.
- The project does not display Company Information, business name, place of registration, registered number, its registered office address and is only intended for the sole purpose school project.
- The system might Display a Privacy Policy informing the user what the business does with the data and that it conforms to the The Data Protection Act.
- The Privacy Policy needs to explain what cookies the website will create and what they are for. (Only for the sole purpose of the school project)
- Have Terms & Conditions, user Data Privacy  Policy pages to show information on how user data is collected