System Design Document
CSCC01 Summer 2021
Team: Seal

Table of Contents

Software Architecture

For this project, we are using a 3-tier architecture setup, with React on the top as our Website, Express running on top of Node in the middle as our server and MongoDB as the bottom database layer.
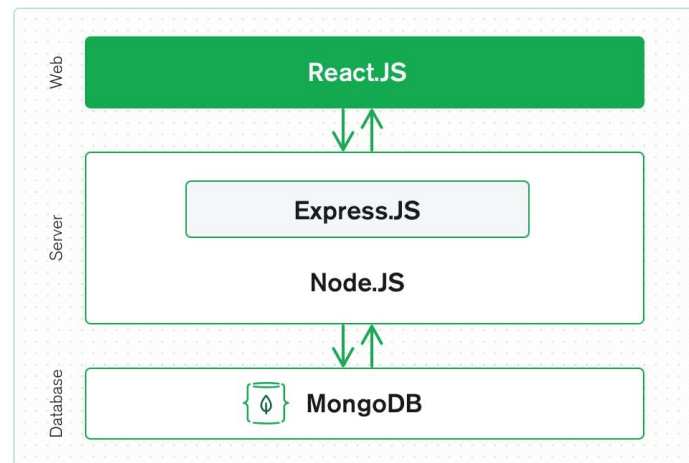
SEAL-3: As a startup founder, I want to create a profile for my company, so that investors can learn about my company.

Class Name: createCompanyProfile
Parent Class: None
Subclasses: None
Responsibilities:
* Allow users input text information regarding building company profile, including company title, tagline, description, and email address.
* Allow users to upload images for the company logo.
* Allow users to include links to external websites, such as Twitter and Facebook, by choice.
* Allow users to include and link members of the company.
* Pass user input into the database.
Collaborators: None

Class Name: editCompanyProfile
Parent Class: None
Subclasses: None
Responsibilities:

* Request field of user input information stored in the database.
* Allow the user to fill in the desired update on their respective company profile.
* Pass user input into the database.
Collaborators: None

Class Name: myCompanyProfile
Parent Class: None
Subclasses: None
Responsibilities:
* Display required information in setting up the company profile. Fields include company title, tagline, description, email address, image upload (logo), external links, and members.
Collaborators: None

Class Name: createCompanyProfile	
Parent Class (if any): None **Subclasses (if any): None**	
Responsibilities: • Allow users input text information regarding building company profile, including company title, tagline, description, and email address. • Allow users to upload images for the company logo. • Allow users to include links to external websites, such as Twitter and Facebook, by choice. • Allow users to include and link members of the company. • Pass user input into the database.	**Collaborators:** •

Class Name: editCompanyProfile	
Parent Class (if any): None **Subclasses (if any): None**	
Responsibilities: • Request field of user input information stored in the database. • Allow the user to fill in the desired update on their respective company profile.	**Collaborators:** •

● Pass user input into the database.	

Class Name: myCompanyProfile	
Parent Class (if any): None **Subclasses (if any): None**	
Responsibilities: ● Display required information in setting up the company profile. Fields include company title, tagline, description, email address, image upload (logo), external links, and members.	**Collaborators:** ●

SEAL-1&2:

- **As an unregistered user, I want to register for an account as a specific role(s), so that I may receive access to certain resources depending on my role(s).**
- **As a registered user, I want to login to my account, so that I can access the resources provided by the community.**

Class Name: User	
Parent Class (if any): None **Subclasses (if any): startupfounder, investor, instructor**	
Responsibilities: ● email ● password ● username ● posts ● comments ● date joined ● role: sf, inv, ins	**Collaborators:** ●

Class Name:Startup Founder

Parent Class (if any):User
Subclasses (if any):

Responsibilities:	Collaborators:
● inherits all from User ● Founder page ● Enrolled courses	●

Class Name:Investor

Parent Class (if any):
Subclasses (if any):

Responsibilities:	Collaborators:
● Investor Page	●

Class Name:Instructor

Parent Class (if any):
Subclasses (if any):

Responsibilities:	Collaborators:
● Courses ● Videos ● Homework Submissions	●

Class Name:Login.js

Parent Class (if any):
Subclasses (if any):

Responsibilities:	Collaborators:
● Take user inputs ● Send request to "/user/login" route for authentication ● Redirect to main post page if authentication success	● users.js(route) ● (Main post page) ● select.js

Class Name:registration.js

Parent Class (if any):

Subclasses (if any):	
Responsibilities: ● Take user input ● Send request to "/user/register" route for creating account ● Selection user role(s) ● Redirect to main post page if account is created	**Collaborators:** ● users.js ● (Main post page) ● select.js

Class Name:select.js	
Parent Class (if any): **Subclasses (if any):**	
Responsibilities: ● Login/Sign up selection	**Collaborators:** ● registration.js ● Login.js

SEAL-7: As a user, I want to watch pre-recorded content uploaded by teachers, so that I can watch them at my own pace.

Class Name: Course	
Parent Class (if any): None **Subclasses (if any):** None	
Responsibilities: ● Containing Information about a specific course ● Also contains the videos associated with said course	**Collaborators:** ● Video

Class Name: Video	
Parent Class (if any): None **Subclasses (if any):** None	
Responsibilities: ● Stores information about the video	**Collaborators:** ●

Class Name: Video Player	
Parent Class (if any): None **Subclasses (if any):** None	
Responsibilities: ● Displays video selected by the user	**Collaborators:** ● Video

SEAL-4: As a user, I want to communicate with other members of the community, so that I can network and connect with other people.

Class Name: Posts	
Parent Class (if any): None **Subclasses (if any): None**	
Responsibilities: ● Display contents of a post to other users. ● Display who created the post. ● Displays a post's tags. ● Allow user to modify the post and send the results to the database. ● Allow user to delete the post and remove the object from the database.	**Collaborators:** ● None

Class Name: Comment	
Parent Class (if any): None **Subclasses (if any): None**	
Responsibilities: ● Displays contents of a comment to other users. ● Displays creator of comment ● Allow user to modify the comment and send the results to the database.	**Collaborators:** ● None

● Allow user to delete the comment and remove the object from the database.	

Class Name: PostViewer

Parent Class (if any): None
Subclasses (if any): None

Responsibilities:	Collaborators:
- Retrieves data from database and configures Posts. - Shows up to some amount of posts on the screen at a time. - Navigate between pages to show more posts. - Redirects user PostViewerSpecifc page when user clicks on post - Add post to database	- Post - PostViewerSpecific - PostWriter

Class Name: PostViewerSpecific

Parent Class (if any): None
Subclasses (if any): None

Responsibilities:	Collaborators:
- Displays the specified Post from PostViewer. - Retrieves, configures, then displays comments from the database. - Displays a specified number of comments and uses a page navigation to switch between all comments. - Adds a comment to this post to the database	- Post - Comment - CommentWriter

Class Name: PostWriter

Parent Class (if any): None
Subclasses (if any): None

Responsibilities:	Collaborators:
- Has textbox to write contents of post - Has list of toggleable tags - Has a submit button to add the post to the database.	None

Class Name: CommentWriter	
Parent Class (if any): None **Subclasses (if any): None**	
Responsibilities: ● Has a textbox to write contents of post ● Has a submit button to add a comment to the database.	**Collaborators:** ● None