System Design

Team:
Coding Lads

Members:
Alex Wong
Han-Shin (Henson) Chen
Hung Jen (Ryan) Lin
Timofey Smetanin
Zining (Jenny) Yu
Simon Ha
Kuan-Te (Henry) Lu

Table of Contents

Frontend CRC:

App
Responsibility: Routing between the main web pages
Collaborator: Sidebar, Landing, SignIn, SignUp, UserDashboard, OrgDashboard, ExploreOpportunities, ManageOpportunities,, ExploreCourses, ManageCourses, Community, Billing, UserContextProvider, PrivateRoute
Parent Class: None
Child Class(es): None

UserContextProvider
Responsibility: Provides context for signed in user's web pages.
Collaborator:
Parent Class: None
Child Class(es): None

PrivateRoute
Responsibility: Routes between signed in user's web pages.
Collaborator: UserContextProvider
Parent Class: None
Child Class(es): None

Sidebar
Responsibility: Display options for users to navigate to different tabs

Collaborator: UserContextConsumer
Parent Class: None
Child Class(es): None

SignUp
Responsibility: Web page for user to sign up
Collaborator: None
Parent Class: None
Child Class(es): None

SignIn
Responsibility: Web page for user to sign in
Collaborator: UserContextProvider
Parent Class: None
Child Class(es): None

Dashboard
Responsibility: Displays calendar events, current courses, suggested events, and news
Collaborator: UserContextConsumer
Parent Class: None
Child Class(es): None

Calender
Responsibility: Displays a calender with user-created events on it

Collaborator: None
Parent Class: None
Child Class(es): None

News
Responsibility: Display any news related to the user
Collaborator: None
Parent Class: None
Child Class(es): None

Workshop
Responsibility: Display workshop events that the user is part of
Collaborator: None
Parent Class: None
Child Class(es): None

YourCourses
Responsibility: Display courses that the user is part of
Collaborator: None
Parent Class: None
Child Class(es): None

ExploreOpportunities

Responsibility: Display listing of available employment, volunteer, and consulting opportunities
Collaborator: OpportunityCard
Parent Class: None
Child Class(es): None

MyOpportunities
Responsibility: Display listing of employment, volunteer, and consulting opportunities that user applied for (if impact learner/consultant) or created (if social initiative)
Collaborator: OpportunityCard
Parent Class: None
Child Class(es): None

OpportunityCard
Responsibility: Display title and description of a particular opportunity
Collaborator: None
Parent Class: None
Child Class(es): None

CreateOpportunity
Responsibility: Page for social initiatives to create a new opportunity.
Collaborator: None
Parent Class: None
Child Class(es): None

ManageOpportunity

Responsibility: Page for social initiatives to manage an existing opportunity.
Collaborator: None
Parent Class: None
Child Class(es): None

ApplyOpportunity
Responsibility: Page for Impact Learners and Impact Consultants to apply for an existing opportunity.
Collaborator: None
Parent Class: None
Child Class(es): None

ExploreCourses
Responsibility: Displays a list of courses that users can enroll in.
Collaborator:
Parent Class: None
Child Class(es): None

MyCourses
Responsibility: Displays a list of enrolled courses that users can click into for more information.
Collaborator:
Parent Class: None
Child Class(es): None

Community
Responsibility: Displays a forum for each course user is enrolled in.

Collaborator:
Parent Class: None
Child Class(es): None

Billing
Responsibility: Displays earnings and invoices.
Collaborator:
Parent Class: None
Child Class(es): None

User	ImpactConsultant, ImpactLearner
• Knows name • Knows email • knows password • knows age	

Interface Role	
• Represent different roles for a user	• User

ImpactLearner	User
• knows name • knows email • knows password • can take courses • can manage applications • can manage events • can manage invoice • can manage social initiatives	• Postings • Course • Application • Event • Invoice • SocialInit

ImpactConsultant	User
• knows name • knows email • knows password • manages courses • manages applications • can view events • manage invoice	• Course • Application • Events • Invoice • Social Initiative

Application	
• knows date	• Posting • impact learner • impact consultant

Posting	
• knows date • knows name • knows description	• impact learner • impact consultant • application • social initiative

Invoice	
• knows User • knows billing information	• impact learner • impact consultant

Course	
• knows name • knows description	• Impact learner • Impact consultant • lectures • social initiative

Lecture	
• Knows lecture times	• Course

Event	
• knows date • knows description • knows name	• impact learner • impact consultant • social initiative

SocialInitiative	
• knows name • knows description	• impact learner • impact consultant • course • posting • event

System Architecture Decomposition

Clients interact with react front-end, each with different components.

Front-end calls exposed back-end, which is broken up into controller/service/DAO layers which are mutually exclusive.

Back-end talks to db to return requested information, then serializes the obj into JSON to pass back to front-end.

Error handling:

All errors with respect to API calls will be dealt with accordingly:

200 if the request is ok and everything happens as it should,

400, if the request is badly formatted or missing information,

405, if the requested user is unauthenticated,

500, if anything else,

From the front-end point of view, will only allow users to do things they are capable of doing.