

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 Cards.....	2
Backend CRC Cards.....	9
System Architecture Design	13
System Design Diagram	14

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

MyCourses
Responsibility: Displays a list of enrolled courses for an impact learner, or a list of instructing courses for an impact consultant.
Collaborator: CourseCard
Parent Class: None
Child Class(es): None

CourseCard
Responsibility: Displays a specific course with its name and status in the form of a card.
Collaborator:
Parent Class: None
Child Class(es): None

CoursePage
Responsibility: Page for accessing all the material for a course, including class sessions, assignments, and quizzes.
Collaborator: CourseSessionRoom
Parent Class: None
Child Class(es): None

CourseSessionRoom
Responsibility: Page for the live class sessions that were created by the instructor of the course
Collaborator:
Parent Class: None
Child Class(es): None

ExploreCourse
Responsibility: Displays a list of courses that an impact learner can enroll in.
Collaborator:
Parent Class: None
Child Class(es): None

CreateCourse
Responsibility: Page for impact consultants to create a new course.
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	
<ul style="list-style-type: none"> • Knows name • Knows email • knows password • knows age 	

Interface	Role
<ul style="list-style-type: none"> • Represent different roles for a user 	<ul style="list-style-type: none"> • User

ImpactLearner	
User	
<ul style="list-style-type: none"> • knows name • knows email • knows password • can take courses • can manage applications • can manage events • can manage invoice • can manage social initiatives 	<ul style="list-style-type: none"> • Postings • Course • Application • Event • Invoice • SocialInit

ImpactConsultant	
User	
<ul style="list-style-type: none"> • knows name • knows email • knows password • manages courses • manages applications • can view events • manage invoice 	<ul style="list-style-type: none"> • Course • Application • Events • Invoice • Social Initiative

Application

- | | |
|--------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------|
| <ul style="list-style-type: none">• knows date | <ul style="list-style-type: none">• Posting• impact learner• impact consultant |
|--------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------|

Posting

- | | |
|---------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------|
| <ul style="list-style-type: none">• knows date• knows name• knows description | <ul style="list-style-type: none">• impact learner• impact consultant• application• social initiative |
|---------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------|

Invoice

- | | |
|--------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------|
| <ul style="list-style-type: none">• knows User• knows billing information | <ul style="list-style-type: none">• impact learner• impact consultant |
|--------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------|

Course

- | | |
|------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <ul style="list-style-type: none">• knows name• knows description | <ul style="list-style-type: none">• Impact learner• Impact consultant• lectures• social initiative• ClassSession |
|------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------|

Lecture

- | | |
|-----------------------------------------------------------------------|----------------------------------------------------------|
| <ul style="list-style-type: none">• Knows lecture times | <ul style="list-style-type: none">• 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

Quiz

- QuizQuestion

QuizQuestion

- Knows question type
- Knows actual question

- Quiz
- QuizOption
- StudentAnswer

QuizOption

- Knows options

- QuizQuestion

Solution

- | | |
|----------------------------------------------------------------------------|----------------------------------------------------------------|
| <ul style="list-style-type: none">• Knows answer to question | <ul style="list-style-type: none">• QuizQuestion |
|----------------------------------------------------------------------------|----------------------------------------------------------------|

ClassSession

- | | |
|-----------------------------------------------------------------------|----------------------------------------------------------|
| <ul style="list-style-type: none">• Knows lecture times | <ul style="list-style-type: none">• Course |
|-----------------------------------------------------------------------|----------------------------------------------------------|

StudentAnswer

- | | |
|------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------|
| <ul style="list-style-type: none">• Knows student's answer to the question | <ul style="list-style-type: none">• ImpactLearner• QuizQuestion |
|------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------|

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.

System Design Diagram

