



Team Yes
CSCC01H3 F - Introduction to Software Engineering
Ilir Dema

Table of Contents

[CRC Cards](#)

[Back End](#)

[Front end](#)

[Signup](#)

[Login](#)

[Profile](#)

[The Zone](#)

[Trivia](#)

[Debate](#)

[Picks And Predictions](#)

[Nav Bar](#)

[General](#)

[Deprecated](#)

[MVC Architecture](#)

CRC Cards

Back End

Acs.js	
Responsibilities <ul style="list-style-type: none">• Handles ACS computations	Collaborators <ul style="list-style-type: none">• DatabaseHelper.js

User.js	
Responsibilities <ul style="list-style-type: none">• Contains personal and administrative information for the account<ul style="list-style-type: none">○ Knows username○ Knows password○ Connected to a Profile○ Knows phone number○ Knows email	Collaborators <ul style="list-style-type: none">• Profile.js

Profile.js	
Responsibilities <ul style="list-style-type: none">• Contains shared information along with app-generated info of the account<ul style="list-style-type: none">○ Knows the profile picture○ Knows about○ Knows status○ Knows ACS○ Knows questionnaire answers○ Has a list of picks○ Has a tracker of other Users○ Knows their debate question○ Knows their debate tier○ Knows how many times user has voted	Collaborators <ul style="list-style-type: none">• User.js

Comment.js

Responsibilities <ul style="list-style-type: none"> • Used to hold a comment for a specific post (stored under their respective post) • Contains information on comments for The Zone <ul style="list-style-type: none"> ○ Knows its object id ○ Knows its username ○ Knows the date it was posted ○ Knows text ○ Knows number of agrees ○ Knows number of disagrees ○ Knows the users who agreed ○ Knows the users who disagreed 	Collaborators

Match.js	
Responsibilities <ul style="list-style-type: none"> • Used for picks and predictions to store into the database • Stores necessary information of a sport match for a daily pick <ul style="list-style-type: none"> ○ Knows team1 ○ Knows team2 ○ Knows date ○ Knows start time ○ Knows end time ○ Knows users and their picks 	Collaborators

Post.js	
Responsibilities <ul style="list-style-type: none"> • Used for storing Posts in the Zone into the database • Stores necessary information for posts <ul style="list-style-type: none"> ○ Knows username ○ Knows the date it was posted ○ Knows content ○ Knows number of agrees ○ Knows number of disagrees ○ Knows the users who agreed ○ Knows the users who 	Collaborators <ul style="list-style-type: none"> • Comment.js

disagreed <ul style="list-style-type: none"> Knows Comments 	
--	--

Question.js	
Responsibilities <ul style="list-style-type: none"> Used in Trivia to store questions into the database Stores necessary information for questions <ul style="list-style-type: none"> Knows the question Knows the correct answer Knows other answers (incorrect) 	Collaborators

Debate.js	
Responsibilities <ul style="list-style-type: none"> Used in Debates to store questions into the database Stores necessary information for debates <ul style="list-style-type: none"> Knows tier Knows question 	Collaborators

DebateController.js	
Responsibilities <ul style="list-style-type: none"> Handles all HTTP requests to the debate module Has logic to parse the HTTP requests Verify data before sending to model 	Collaborators <ul style="list-style-type: none"> DatabaseCreate.js DatabaseRead.js DatabaseUpdate.js DatabaseDelete.js

LoginController.js	
Responsibilities <ul style="list-style-type: none"> Handles all HTTP requests to the 	Collaborators <ul style="list-style-type: none"> DatabaseCreate.js

login module <ul style="list-style-type: none"> • Has logic to parse the HTTP requests • Verify data before sending to model to authenticate user 	<ul style="list-style-type: none"> • DatabaseRead.js • DatabaseUpdate.js • DatabaseDelete.js
---	---

TheZoneController.js	
Responsibilities <ul style="list-style-type: none"> • Handles all HTTP requests to the Zone module • Has logic to parse the HTTP requests • Verify data before sending to model 	Collaborators <ul style="list-style-type: none"> • DatabaseCreate.js • DatabaseRead.js • DatabaseUpdate.js • DatabaseDelete.js

UserController.js	
Responsibilities <ul style="list-style-type: none"> • Handles all HTTP requests to the user module • Has logic to parse the HTTP requests • Verify data before sending to model 	Collaborators <ul style="list-style-type: none"> • DatabaseCreate.js • DatabaseRead.js • DatabaseUpdate.js • DatabaseDelete.js

ProfileController.js	
Responsibilities <ul style="list-style-type: none"> • Handles all HTTP requests to the profile module • Has logic to parse the HTTP requests • Verify data before sending to model 	Collaborators <ul style="list-style-type: none"> • DatabaseCreate.js • DatabaseRead.js • DatabaseUpdate.js • DatabaseDelete.js

SignupController.js	
Responsibilities <ul style="list-style-type: none"> • Handles all HTTP requests to the profile module • Has logic to parse the HTTP requests • Verify data before sending to model 	Collaborators <ul style="list-style-type: none"> • DatabaseCreate.js • DatabaseRead.js • DatabaseUpdate.js • DatabaseDelete.js

TriviaController.js

Responsibilities <ul style="list-style-type: none"> • Handles all HTTP requests to the trivia module • Has logic to parse the HTTP requests • Verify data before sending to model 	Collaborators <ul style="list-style-type: none"> • DatabaseCreate.js • DatabaseRead.js • DatabaseUpdate.js • DatabaseDelete.js
--	--

PicksPredictionsController.js	
Responsibilities <ul style="list-style-type: none"> • Handles all HTTP requests to the picks and predictions module • Has logic to parse the HTTP requests • Verify data before sending to model 	Collaborators <ul style="list-style-type: none"> • DatabaseCreate.js • DatabaseRead.js • DatabaseUpdate.js • DatabaseDelete.js

DatabaseCreate.js	
Responsibilities <ul style="list-style-type: none"> • Handles any request to create documents in the database. 	Collaborators <ul style="list-style-type: none"> • DatabaseRead.js • DatabaseUpdate.js • DatabaseHelper.js • User.js

DatabaseRead.js	
Responsibilities <ul style="list-style-type: none"> • Handles any request to read documents in the database. 	Collaborators <ul style="list-style-type: none"> • DatabaseHelper.js

DatabaseUpdate.js	
Responsibilities <ul style="list-style-type: none"> • Handles any request to update existing documents in the database. 	Collaborators <ul style="list-style-type: none"> • DatabaseRead.js • DatabaseDelete.js • DatabaseHelper.js

DatabaseDelete.js	
Responsibilities <ul style="list-style-type: none"> • Handles any request to delete existing documents in the database. 	Collaborators

DatabaseHelper.js	
Responsibilities <ul style="list-style-type: none"> Helps the other Database classes by providing constants. 	Collaborators

StatusMessages.js	
Responsibilities <ul style="list-style-type: none"> Contains status messages for responses the controller classes will use. 	Collaborators

Front end

App.jsx	
Responsibilities <ul style="list-style-type: none"> Display different pages depending on state Tracks current user logged in 	Collaborators <ul style="list-style-type: none"> SignupPage.jsx LoginPage.jsx ProfilePage.jsx OpenCourtPage.jsx TheZonePage.jsx TriviaPage.jsx DebatePage.jsx PicksAndPredictionsPage.jsx TopNavBar.jsx

index.js	
Responsibilities <ul style="list-style-type: none"> Renders the App component (topmost level container of the application) into the html page that will be shown to the user 	Collaborators <ul style="list-style-type: none"> App.jsx

HttpClient.jsx	
Responsibilities <ul style="list-style-type: none"> Holds common information necessary for front to back end connection 	Collaborators

Note: reportWebVitals.js, setupTests.js were auto-generated files from Create React App and are separate from the system design.

Signup

SignupCalls.js	
Responsibilities <ul style="list-style-type: none">• Send sign up information to backend server using HTTP to create a new account• Check if a username, phone, or email already exists	Collaborators <ul style="list-style-type: none">• HttpClient.jsx

SignupPage.jsx	
Responsibilities <ul style="list-style-type: none">• Display the signup form• Send sign up information to SignupCalls to create a new account	Collaborators <ul style="list-style-type: none">• SignupCalls.js

Login

LoginCalls.js	
Responsibilities <ul style="list-style-type: none">• Call the backend server to login and authenticate user	Collaborators <ul style="list-style-type: none">• HttpClient.jsx

LoginPage.jsx	
Responsibilities <ul style="list-style-type: none">• Display the login form• Sends the login details to the LoginCalls to log the user in	Collaborators <ul style="list-style-type: none">• LoginCalls.js

Profile

ProfileCalls.js	
Responsibilities <ul style="list-style-type: none"> • Call the backend server to access the profile information of wanted user(s) • Call the backend server to edit profile information of the currently logged in user 	Collaborators <ul style="list-style-type: none"> • HttpClient.jsx

ProfilePage.jsx	
Responsibilities <ul style="list-style-type: none"> • Display the components in Profile • Know all information regarding a users profile as well as the current user • Give profile information to appropriate children • Call fetch to retrieve and send info in server • Receives and controls input from collaborators 	Collaborators <ul style="list-style-type: none"> • ProfileCalls.js • ProfilePicture.jsx • ProfileAbout.jsx • ProfileStatus.jsx • ProfileACS.jsx • ProfileSocial.jsx • ProfileRadar.jsx

ProfilePicture.jsx	
Responsibilities <ul style="list-style-type: none"> • Display the profile picture • Upload a profile picture from the user's given link • Save changes to the profile picture 	Collaborators <ul style="list-style-type: none"> • ProfilePage.jsx

ProfileAbout.jsx	
Responsibilities <ul style="list-style-type: none"> • Display the about section in profile • Edit the about section • Save changes to the profile about 	Collaborators <ul style="list-style-type: none"> • ProfilePage.jsx

ProfileStatus.jsx	
Responsibilities <ul style="list-style-type: none"> • Display the profile status 	Collaborators <ul style="list-style-type: none"> • ProfilePage.jsx

<ul style="list-style-type: none"> • Edit the profile status • Save change to the profile status 	
--	--

ProfileACS.jsx	
Responsibilities <ul style="list-style-type: none"> • Display the ACS of the wanted user 	Collaborators <ul style="list-style-type: none"> • ProfilePage.jsx

ProfileSocial.jsx	
Responsibilities <ul style="list-style-type: none"> • Save links to social media of user • Display social media links of wanted user 	Collaborators <ul style="list-style-type: none"> • ProfilePage.jsx

ProfileRadar.jsx	
Responsibilities <ul style="list-style-type: none"> • Have button to allow follow/unfollow • Display a list of followed users with acs for a given users • Access other profiles on followed list 	Collaborators <ul style="list-style-type: none"> • ProfilePage.jsx

The Zone

TheZoneCalls.js	
Responsibilities <ul style="list-style-type: none"> • Manages getting and sending posts datas from backend 	Collaborators <ul style="list-style-type: none"> • HttpClient.jsx

TheZonePage.jsx	
Responsibilities <ul style="list-style-type: none"> • Display the components for the zone 	Collaborators <ul style="list-style-type: none"> • TheZone.jsx

--	--

TheZone.jsx	
Responsibilities <ul style="list-style-type: none"> Provides navigation to the other pages of the app Display posts of users 	Collaborators <ul style="list-style-type: none"> TheZonePostList.jsx

TheZoneCreatePost.jsx	
Responsibilities <ul style="list-style-type: none"> Display the editor to create a Post Send input to TheZoneCalls 	Collaborators <ul style="list-style-type: none"> TheZoneCalls.js

TheZonePost.jsx	
Responsibilities <ul style="list-style-type: none"> Display the post components Display the content of the post Get content of the post from the back end with TheZoneCalls Send input to back end with TheZoneCalls 	Collaborators <ul style="list-style-type: none"> TheZoneCreateComment.jsx TheZoneComment.jsx TheZoneCalls.js TheZoneAgree.jsx AccessProfileText.jsx

TheZonePostList.jsx	
Responsibilities <ul style="list-style-type: none"> Get list of posts to display from the backend Add a post to front end if a post is created in the back end Remove a post in the front end if it is removed from the back end 	Collaborators <ul style="list-style-type: none"> TheZoneCalls.js TheZoneCreatePost.jsx

TheZoneCreateComment.jsx	
Responsibilities <ul style="list-style-type: none"> • Display the editor to create a comment • Send input to TheZoneCalls 	Collaborators <ul style="list-style-type: none"> • TheZoneCalls.js

TheZoneComment.jsx	
Responsibilities <ul style="list-style-type: none"> • Display comments to a post • Get comments to a post from TheZoneCalls 	Collaborators <ul style="list-style-type: none"> • TheZoneCalls.js • TheZoneAgree.jsx • AccessProfileText.jsx

TheZoneAgree.jsx	
Responsibilities <ul style="list-style-type: none"> • Display votes of a post/comment • Get votes for a post/comment from TheZoneCalls • Display buttons to vote on a post/comment • Submit and update votes for a post/comment with TheZoneCalls 	Collaborators <ul style="list-style-type: none"> • TheZoneCalls.js

Trivia

TriviaCalls.js	
Responsibilities <ul style="list-style-type: none"> • Makes HTTP requests to the backend for Trivia related data <ul style="list-style-type: none"> ◦ Get 10 trivia questions ◦ Get an answer to a trivia question ◦ Update ACS based on a player's answers to the 10 trivia questions and return the new ACS 	Collaborators <ul style="list-style-type: none"> • HttpClient.jsx

<ul style="list-style-type: none"> ○ Create a trivia question and store in backend ○ Read all trivia questions in the backend ○ Update a trivia question in the backend ○ Delete a trivia question in the backend 	
---	--

TriviaPage.jsx	
Responsibilities <ul style="list-style-type: none"> ● The container for displaying the Trivia game <ul style="list-style-type: none"> ○ Mostly the CSS for page dimensions and alignment of elements 	Collaborators <ul style="list-style-type: none"> ● Trivia.jsx

Trivia.jsx	
Responsibilities <ul style="list-style-type: none"> ● Manages game state and which screen of the game to show according to the state <ul style="list-style-type: none"> ○ In other words, the container for all the smaller Trivia components and controls their props ● Uses TriviaCalls to load/send Trivia data back to the backend to keep it updated on the current game state 	Collaborators <ul style="list-style-type: none"> ● TriviaStart.jsx ● LoadingScreen.jsx ● TriviaGame.jsx ● TriviaResults.jsx ● TriviaCalls.js

TriviaStart.jsx	
Responsibilities <ul style="list-style-type: none"> ● The main menu / start screen of Trivia <ul style="list-style-type: none"> ○ Gives user the option to choose between a Solo game or Head to Head game ○ Optionally display any errors that prevent the game from continuing normally 	Collaborators

TriviaGame.jsx	
Responsibilities <ul style="list-style-type: none"> • Displays each Trivia question in sequence, providing how much time there is for previewing a question and answering one • Records the player's answer to each Trivia question 	Collaborators <ul style="list-style-type: none"> • TriviaQuestion.jsx

TriviaQuestion.jsx	
Responsibilities <ul style="list-style-type: none"> • Displays a single Trivia question and their answers so a player can answer a Trivia question <ul style="list-style-type: none"> ◦ Randomizes the order of the answers using functionality provided from utilities • Manages the timers for previewing a question and selecting an answer • Communicates whether the player selected a correct or incorrect answer 	Collaborators <ul style="list-style-type: none"> • utilities.js

TriviaResults.jsx	
Responsibilities <ul style="list-style-type: none"> • Displays the results of a Trivia game <ul style="list-style-type: none"> ◦ Number of correct answers out of the total number of questions ◦ The updated ACS • Lets the player Play Again • Lets the player go back to the Main Menu / Start 	Collaborators

Debate

DebateCalls.js	
Responsibilities	Collaborators

<ul style="list-style-type: none"> Manages sending and receiving questions and answers for Debate 	<ul style="list-style-type: none"> HttpClient.jsx
--	--

DebatePage.jsx	
Responsibilities <ul style="list-style-type: none"> Display the components in Debate 	Collaborators <ul style="list-style-type: none"> Debate.jsx

Debate.jsx	
Responsibilities <ul style="list-style-type: none"> Get the current topic for the debate for the user from DebateCalls Display the list of posts 	Collaborators <ul style="list-style-type: none"> DebatePostList.jsx DebateCalls.jsx

DebateCreatePost.jsx	
Responsibilities <ul style="list-style-type: none"> Display the editor to create a post for debate Get the current debate topic from DebateCalls Send input to DebateCalls 	Collaborators <ul style="list-style-type: none"> DebateCalls.jsx

DebatePost.jsx	
Responsibilities <ul style="list-style-type: none"> Display the debate post components Get the datas for the answer to a debate from DebateCalls 	Collaborators <ul style="list-style-type: none"> DebateCalls.jsx DebateAgree.jsx

DebateAgree.jsx	
Responsibilities <ul style="list-style-type: none"> Display percentage of people agree with the post Get percentage of people agree with the post from DebateCalls Get ACS gained from debate post from DebateCalls Display ACS gained from debate post 	Collaborators <ul style="list-style-type: none"> DebateCalls.jsx

<ul style="list-style-type: none"> • Display slider from 0-100 and button to agree on a scale • Send user input to DebateCalls and update the agree percentage on front end 	
---	--

DebatePostList.jsx	
Responsibilities <ul style="list-style-type: none"> • Display the list of posts in a debate • Get group of posts to display from back-end with DebateCalls • Add a post to the list when post is successfully created by the user 	Collaborators <ul style="list-style-type: none"> • DebateCalls.jsx • DebatePostGroup.jsx • DebateCreatePost.jsx

DebatePostGroup.jsx	
Responsibilities <ul style="list-style-type: none"> • Display the list of posts in the group and a divider between the groups 	Collaborators <ul style="list-style-type: none"> • DebatePost.jsx

Picks And Predictions

PicksCalls.js	
Responsibilities <ul style="list-style-type: none"> • Manages sending and receiving picks and predictions of current user 	Collaborators <ul style="list-style-type: none"> • HttpClient.jsx

PicksAndPredictionsPage.jsx	
Responsibilities <ul style="list-style-type: none"> • Display the components in Picks and Predictions 	Collaborators <ul style="list-style-type: none"> • DailyPicks.jsx • PreseasonPicks.jsx • PlayOffPicks.jsx

DailyPicks.jsx	
Responsibilities	Collaborators

<ul style="list-style-type: none"> • Display the list of daily picks • Get the list of matches for the day from PicksCalls 	<ul style="list-style-type: none"> • PicksCalls.js • SinglePick.jsx
--	---

SinglePick.jsx	
Responsibilities <ul style="list-style-type: none"> • Display the match user is to pick from <ul style="list-style-type: none"> ◦ Shows date & time of match ◦ Teams that are playing • Allow user to choose their prediction of match winner before the match starts • Show match ongoing during the time the match is scheduled • Show the winner of the match after the game is complete 	Collaborators

PreseasonPicks.jsx	
Responsibilities <ul style="list-style-type: none"> • Display the components for preseason picks during preseason • Sends user input to PicksCalls to keep track of player picks • Get data from PicksCalls to display NBA preseason data for picks 	Collaborators <ul style="list-style-type: none"> • PicksCalls.js

PlayoffPicks.jsx	
Responsibilities <ul style="list-style-type: none"> • Display the components for playoff during NBA playoff • Sends user input to PicksCalls to keep track of player picks 	Collaborators <ul style="list-style-type: none"> • PicksCalls.js • PlayoffBracket.jsx

PlayoffBracket.jsx	
Responsibilities <ul style="list-style-type: none"> • Display the playoff bracket during 	Collaborators <ul style="list-style-type: none"> • PicksCalls.js

NBA playoff <ul style="list-style-type: none"> • Sends user input to PicksCalls to keep track of bracket picks • Get results of the playoff bracket form PicksCalls 	
---	--

Nav Bar

TopNavBar.jsx	
Responsibilities <ul style="list-style-type: none"> • Display the logo • Display the button to go back to the zone • Display the current user logged in 	Collaborators <ul style="list-style-type: none"> • CurrentUser.jsx

CurrentUser.jsx	
Responsibilities <ul style="list-style-type: none"> • Display the current user's name and picture • Display a dropdown menu when clicked 	Collaborators <ul style="list-style-type: none"> • DropdownButton.jsx

DropdownButton.jsx	
Responsibilities <ul style="list-style-type: none"> • Display button to redirect to profile page • Display button to log out 	Collaborators

General

AccessProfile.jsx	
Responsibilities <ul style="list-style-type: none"> • Allows users to be redirected to a certain profile by clicking their profile picture 	Collaborators <ul style="list-style-type: none"> • App.jsx

AccessProfileText.jsx	
Responsibilities <ul style="list-style-type: none"> Allows users to be redirected to a certain profile by clicking their name 	Collaborators <ul style="list-style-type: none"> App.jsx

LoadingScreen.jsx	
Responsibilities <ul style="list-style-type: none"> Shows a loading screen containing descriptive text and a loading spinner 	Collaborators

utilities.js	
Responsibilities <ul style="list-style-type: none"> Provides miscellaneous functions for manipulating data used in the front end <ul style="list-style-type: none"> Shuffling an array 	Collaborators

Deprecated

These components are no longer used but they are still in the repository since they weren't removed in time.

OpenCourtPage.jsx	
Responsibilities	Collaborators

It's purpose was moved to The Zone.

ScoreTicker.jsx	
Responsibilities <ul style="list-style-type: none"> Displaying Live Game data and notifications for the user Get data from the backend about live game data to display 	Collaborators <ul style="list-style-type: none"> TheZoneCalls.jsx

TheZoneAgreeScale.jsx	
Responsibilities <ul style="list-style-type: none"> • Displays a scale from 0-100 to agree/disagree with a post and a button to submit • Send user input for agree percentage to the back end with TheZoneCalls • Get new agree percentage with TheZoneCalls to display on the front-end 	Collaborators <ul style="list-style-type: none"> • TheZoneCalls.jsx

Note: Refactored into DebateAgree

MVC Architecture

