

System Design - SPORTCRED

The Code Manglers

Renzo Reyes (reyesren), Ian Gu (guian), Victor Guo (guovicto), Kevin Oh (ohkevin2), Farhan Chowdhury (chowd601)

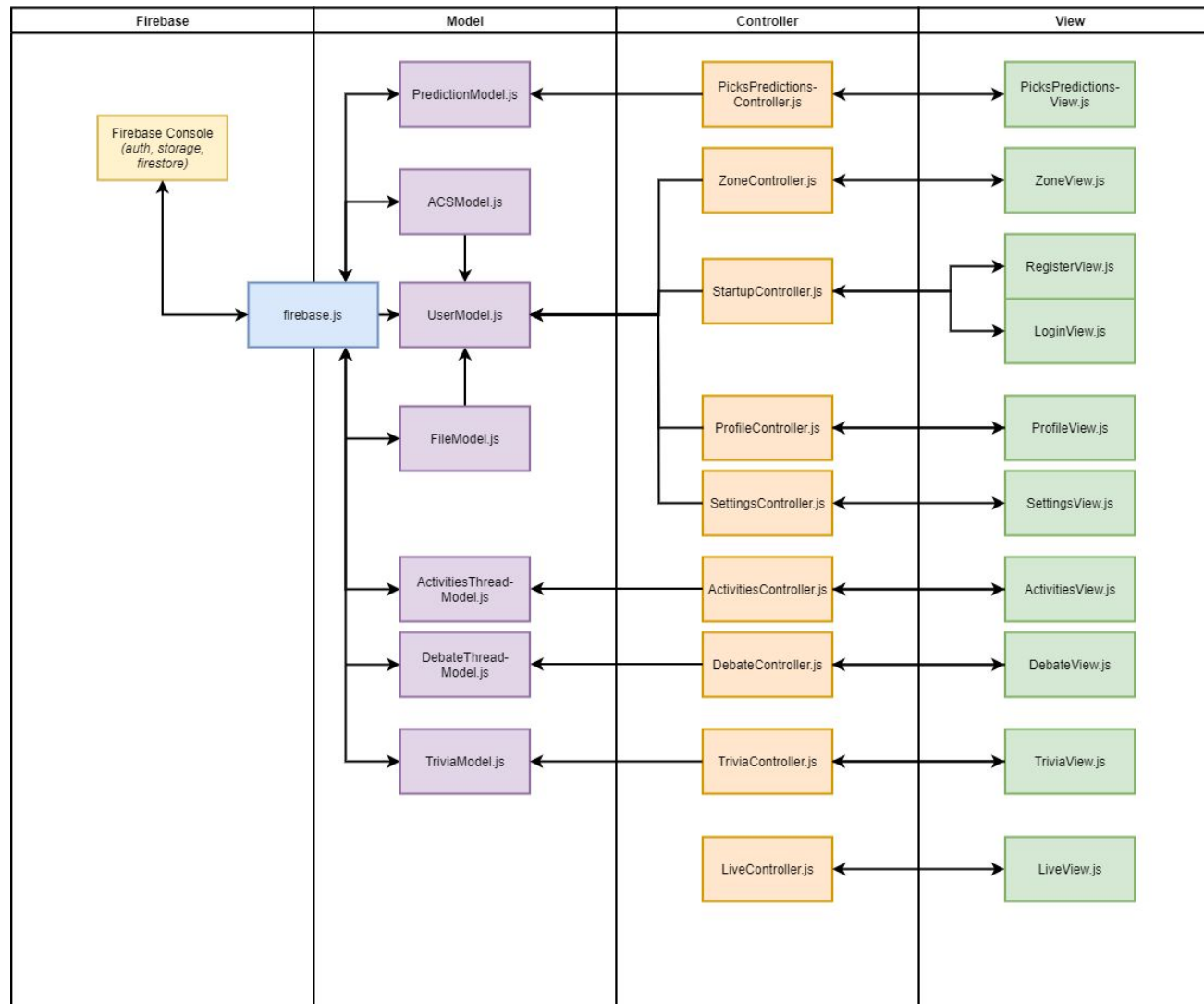
Course: CSC C01 @ University of Toronto Scarborough

Professor: Illir Dema

Table of Contents

Architecture Diagram	3
Structure	4
Models	6
Controllers	10
Views	18

Architecture Diagram



Structure

Class name:	App.js
Parent class:	None
Subclasses:	None
Responsibilities :	<ul style="list-style-type: none">• This is the entry point of the application. It serves the navigation component.
Collaborators:	<ul style="list-style-type: none">• AppNavigator.js

Class name:	AppNavigator.js
Parent class:	None
Subclasses:	None
Responsibilities :	<ul style="list-style-type: none">• AppNavigator.js holds all the controllers, and allows the user to switch between each page.
Collaborators:	All controllers

Class name:	Firebase.js
-------------	-------------

Parent class:	None
Subclasses:	
Responsibilities :	<ul style="list-style-type: none">● Firebase.js provides the basic connection and configuration for using the firebase service
Collaborators:	All models

Models

Class name:	UserModel.js
Parent class:	None
Subclasses:	None
Responsibilities :	<ul style="list-style-type: none">• This class handles any requests to Firebase related to User Information (e.g. Username, Password, Custom ACS score, etc.)
Collaborators:	<ul style="list-style-type: none">• Firebase.js• FileModel.js

Class name:	ACSMModel.js
Parent class:	None
Subclasses:	None
Responsibilities :	<ul style="list-style-type: none">• This class handles any requests to Firebase related to a User's ACS score
Collaborators:	<ul style="list-style-type: none">• UserModel.js• Firebase.js

Class name:	PredictionModel.js
Parent class:	None
Subclasses:	None
Responsibilities :	<ul style="list-style-type: none"> ● This class handles requests to Firebase related to a User's picks and predictions
Collaborators:	<ul style="list-style-type: none"> ● UserModel.js ● Firebase.js

Class name:	PostModel.js
Parent class:	None
Subclasses:	None
Responsibilities :	<ul style="list-style-type: none"> ● This class handles requests to Firebase related to an activity thread.
Collaborators:	<ul style="list-style-type: none"> ● UserModel.js ● Firebase.js

Class name:	DebateThreadModel.js
Parent class:	None

Subclasses:	None
Responsibilities :	<ul style="list-style-type: none"> • This class handles requests to Firebase related to a debate thread.js
Collaborators:	<ul style="list-style-type: none"> • UserModel.js • Firebase.js

Class name:	TriviaModel.js
Parent class:	None
Subclasses:	None
Responsibilities :	<ul style="list-style-type: none"> • This class handles requests to Firebase related to trivia questions and answers, and user history
Collaborators:	<ul style="list-style-type: none"> • UserModel.js • Firebase.js

Class name:	TriviaChallengeModel.js
Parent class:	None

Subclasses:	None
Responsibilities :	<ul style="list-style-type: none">● This class handles requests to Firebase related to user challenges and history
Collaborators:	<ul style="list-style-type: none">● UserModel.js● Firebase.js

Controllers

Class name:	PicksPredictionsController.js
Parent class:	None
Subclasses:	None
Responsibilities :	<ul style="list-style-type: none">• The class will get the necessary teams and brackets to provide to the view• The class will retrieve the picks and predictions from the view, to then send it to the PredictionModel
Collaborators:	<ul style="list-style-type: none">• PicksPredictionsView.js• PredictionModel.js

Class name:	ZoneController.js
Parent class:	None
Subclasses:	List all the subclasses separated by a comma.
Responsibilities :	<ul style="list-style-type: none">• The class will listen for user to choose which part of the app to navigate to while on ZoneView
Collaborators:	<ul style="list-style-type: none">• ZoneView.js• UserModel.js

Class name:	StartupController.js
Parent class:	None
Subclasses:	None
Responsibilities :	<ul style="list-style-type: none"> • The class will send login information to the UserModel for login • The class will send registration information to the UserModel for registration • The class will provide the necessary information to either RegisterView or LoginView for display
Collaborators:	<ul style="list-style-type: none"> • RegisterView.js • LoginView.js • UserModel.js

Class name:	ProfileController.js
Parent class:	None
Subclasses:	None
Responsibilities :	<ul style="list-style-type: none"> • The class will get the user's information from the UserModel to then populate the view of the profile • The class will listen for profile edits, and update the

	UserModel of those edits the user has made.
Collaborators:	<ul style="list-style-type: none"> ● ProfileView.js ● UserModel.js

Class name:	SettingsController.js
Parent class:	None
Subclasses:	None
Responsibilities :	<ul style="list-style-type: none"> ● The class will retrieve the user's custom settings that they want from the app ● The class will get any updated user settings and update the UserModel of these settings
Collaborators:	<ul style="list-style-type: none"> ● SettingsView.js ● UserModel.js

Class name:	ActivitiesController.js
Parent class:	None
Subclasses:	None
Responsibilities :	<ul style="list-style-type: none"> ● The class allows users to contribute to discussions through ActivitiesThreadModel

Collaborators:	<ul style="list-style-type: none"> ● ProfileView.js ● ThreadModel.js
----------------	--

Class name:	DebateController.js
Parent class:	None
Subclasses:	None
Responsibilities :	<ul style="list-style-type: none"> ● This class allows users to contribute to discussions through DebateThreadModel
Collaborators:	<ul style="list-style-type: none"> ● DebateView.js ● ThreadModel.js

Class name:	TriviaController.js
Parent class:	None
Subclasses:	None
Responsibilities :	<ul style="list-style-type: none"> ● This class feeds a “trivia game” to the view ● It will listen for user input and write to TriviaModel accordingly
Collaborators:	<ul style="list-style-type: none"> ● TriviaView.js ● TriviaModel.js

Class name:	TriviaHeadToHeadController.js
Parent class:	None
Subclasses:	None
Responsibilities :	<ul style="list-style-type: none"> • This class displays the tabs for a user to challenge another user, their incoming challenges, and their history.
Collaborators:	<ul style="list-style-type: none"> • ChallengeUsersView.js • IncomingChallengesView.js • CompletedChallengesView.js • TriviaChallengeModel.js • UserModel.js • TriviaModel.js

Class name:	TriviaStartGameController.js
Parent class:	None
Subclasses:	None
Responsibilities	<ul style="list-style-type: none"> • Navigate to the loading screen when user presses start game
Collaborators	<ul style="list-style-type: none"> • TriviaLoadingScreenView

Class name:	TriviaLoadingScreenController.js
Parent class:	None
Subclasses:	None
Responsibilities	<ul style="list-style-type: none"> ● Show a loading screen until data is retrieved from the database ● Route to trivia game after loading is done
Collaborators	<ul style="list-style-type: none"> ● TriviaLoadingScreenView ● TriviaModel

Class name:	TriviaMainGameController.js
Parent class:	None
Subclasses:	None
Responsibilities	<ul style="list-style-type: none"> ● Send information to the main trivia game view so the game can start ● End the game and navigate to the results page
Collaborators	<ul style="list-style-type: none"> ● TriviaMainGameView

Class name:	TriviaResultsController.js
-------------	----------------------------

Parent class:	None
Subclasses:	None
Responsibilities	<ul style="list-style-type: none"> • Send results to the results view and render it • Record the results in the database • Navigate to the home page after user presses continue
Collaborators	<ul style="list-style-type: none"> • TriviaResultsView • TriviaModel

Class name:	LiveController.js
Parent class:	None
Subclasses:	None
Responsibilities :	<ul style="list-style-type: none"> • This class is a placeholder for a future feature
Collaborators:	<ul style="list-style-type: none"> • LiveView.js

Class name:	CreatePostController.js
Parent class:	None
Subclasses:	None
Responsibilities :	<ul style="list-style-type: none"> • This class is responsible for processing data from CreatePostView and communicating with the database to retrieve/add data
Collaborators:	<ul style="list-style-type: none"> • CreatePostView.js • Firebase.js • UserModel.js • PostModel.js

Class name:	FullPostController.js
Parent class:	None
Subclasses:	None
Responsibilities :	<ul style="list-style-type: none"> • This class is responsible for retrieving all post data from the backend and sending it to be displayed
Collaborators:	<ul style="list-style-type: none"> • FullPostView.js • Firebase.js • PostModel.js

Class name:	TheZoneController.js
Parent class:	None
Subclasses:	None
Responsibilities :	<ul style="list-style-type: none"> • This class is responsible for retrieving post data from the database to display in TheZoneView.js
Collaborators:	<ul style="list-style-type: none"> • TheZoneView.js • Firebase.js

Views

Class name:	PicksPredictionsView.js
Parent class:	None
Subclasses:	None
Responsibilities :	<ul style="list-style-type: none"> • represent the UI components for Picks & Predictions in the application

Collaborators:	<ul style="list-style-type: none"> PicksPredictionsController.js
----------------	---

Class name:	RegisterView.js
Parent class:	None
Subclasses:	List all the subclasses separated by a comma.
Responsibilities :	<ul style="list-style-type: none"> the UI component for Register in the application
Collaborators:	<ul style="list-style-type: none"> StartupController.js

Class name:	LoginView.js
Parent class:	None
Subclasses:	None.
Responsibilities :	<ul style="list-style-type: none"> the UI component for Login in the application
Collaborators:	<ul style="list-style-type: none"> LoginController.js

Class name:	ProfileView.js
Parent class:	None
Subclasses:	None
Responsibilities :	<ul style="list-style-type: none"> the UI component for Profile in the application
Collaborators:	<ul style="list-style-type: none"> ProfileController.js

Class name:	SettingsView.js
Parent class:	None
Subclasses:	None
Responsibilities :	<ul style="list-style-type: none"> the UI component for Settings in the application
Collaborators:	<ul style="list-style-type: none"> SettingsController.js

Class name:	ActivitiesView.js
Parent class:	None

Subclasses:	None
Responsibilities :	<ul style="list-style-type: none"> the UI component for Activities in the application
Collaborators:	<ul style="list-style-type: none"> ActivitiesController.js

Class name:	DebateView.js
Parent class:	None
Subclasses:	None
Responsibilities :	<ul style="list-style-type: none"> the UI component for Debate in the application
Collaborators:	<ul style="list-style-type: none"> DebateController.js

Class name:	TriviaView.js
Parent class:	None
Subclasses:	None
Responsibilities :	<ul style="list-style-type: none"> the UI component for Trivia in the application

Collaborators:	<ul style="list-style-type: none"> ● TriviaController.js
----------------	---

Class name:	TriviaLandingView.js
Parent class:	None
Subclasses:	None
Responsibilities :	<ul style="list-style-type: none"> ● The landing view for Trivia in the application
Collaborators:	<ul style="list-style-type: none"> ● TriviaController.js

Class name:	TriviaStartGameView.js
Parent class:	None
Subclasses:	None
Responsibilities	<ul style="list-style-type: none"> ● Show a start game button the user
Collaborators:	<ul style="list-style-type: none"> ● TriviaStartGameController

Class name:	TriviaLoadingScreenView.js
Parent class:	None
Subclasses:	None
Responsibilities	<ul style="list-style-type: none"> ● Show a loading screen to the user
Collaborators:	<ul style="list-style-type: none"> ● TriviaLoadingScreenController

Class name:	TriviaMainGameView.js
Parent class:	None
Subclasses:	None
Responsibilities	<ul style="list-style-type: none"> ● Show the main trivia game to the user ● Show question count ● Show score ● Show question ● Show options ● Show if user pressed correct answer or wrong answer

Collaborators:	<ul style="list-style-type: none"> ● TriviaMainGameController
----------------	--

Class name:	TriviaResultsView.js
Parent class:	None
Subclasses:	None
Responsibilities	<ul style="list-style-type: none"> ● Show results view to the user ● Show final score to the user ● Show ACS change to the user
Collaborators:	<ul style="list-style-type: none"> ● TriviaResultsController

Class name:	ChallengeUsersView.js
Parent class:	None
Subclasses:	None
Responsibilities :	<ul style="list-style-type: none"> ● the UI component for users to challenge another user to trivia
Collaborators:	<ul style="list-style-type: none"> ● TriviaHeadToHeadController.js

Class name:	IncomingChallengesView.js
-------------	---------------------------

Parent class:	None
Subclasses:	None
Responsibilities :	<ul style="list-style-type: none"> the UI component to view a users incoming challenges
Collaborators:	<ul style="list-style-type: none"> TriviaHeadToHeadController.js

Class name:	CompletedChallengesView.js
Parent class:	None
Subclasses:	None
Responsibilities :	<ul style="list-style-type: none"> the UI component for a user to view their completed challenges
Collaborators:	<ul style="list-style-type: none"> TriviaHeadToHeadController.js

Class name:	LiveView.js
Parent class:	None

Subclasses:	None
Responsibilities :	<ul style="list-style-type: none"> the UI component for Live View in the application
Collaborators:	<ul style="list-style-type: none"> LiveController.js

Class name:	CreatePostView.js
Parent class:	None
Subclasses:	None
Responsibilities :	<ul style="list-style-type: none"> the UI component for creating posts
Collaborators:	<ul style="list-style-type: none"> CreatePostController.js

Class name:	FullPostView.js
Parent class:	None
Subclasses:	None
Responsibilities :	<ul style="list-style-type: none"> the UI component for viewing full posts

Collaborators:	<ul style="list-style-type: none"> ● FullPostController.js
----------------	---

Class name:	TheZoneView.js
Parent class:	None
Subclasses:	None
Responsibilities :	<ul style="list-style-type: none"> ● the UI component for TheZone (should list all relevant posts user would see and allow them to go to create post and navigate to other pages)
Collaborators:	<ul style="list-style-type: none"> ● TheZoneController.js