

Creative Ideation

Software Design Document

By: Kellen Evoy, Matthew Marini, Vanessa Li

Table of Contents

1 Introduction.....	3
2 Software Architecture	4
2.1 Software System Architecture	4
3 Detailed Design.....	5
3.1 User Interface Layer.....	17
3.1.1 User and Group Management Overview Diagram	18
3.1.2 Session Management Overview Diagram	18
3.1.3 Artificial Assistance Overview Diagram.....	19
3.2 Application Logic Layer	19
3.2.1 UserViewModel Overview.....	20
3.2.2 TeamViewModel Overview	20
3.2.3 GroupViewModel Overview.....	21
3.2.4 SessionViewModel Overview	21
3.2.5 SessionItemViewModel Overview.....	21
3.2.6 SessionSettingsViewModel Overview	22
3.2.7 ChatbotViewModel Overview	22
3.3 Domain Layer	22
3.3.1 User Information Overview	23
3.3.2 Team Information Overview	23
3.3.3 Group Information Overview	24
3.3.4 Session Information Overview	24
3.3.5 Session Item Information Overview	24
3.3.6 Session Settings Information Overview.....	25
3.4 Infrastructure Layer	25
3.4.1 APIs Overview.....	25
3.4.2 Data Persistence Overview.....	26
4 Interaction Model.....	15
4.1 Manage Profile.....	26
4.1.1 Behavioural Model - Manage Profile	27
4.1.2 Collaboration Overview-Manage Profile	16
4.1.3 Structural Realization Manage Groups.....	17
4.2 Manage Teams.....	17
4.2.1 Behavioural Model - Manage Teams.....	18
4.2.2 Collaboration Overview - Manage Teams	18
4.2.3 Structural Realization Manage Groups.....	19
4.3 Manage Groups.....	19
4.3.1 Behavioural Model-Manage Groups	20
4.3.2 Collaboration Overview - Manage Teams	20

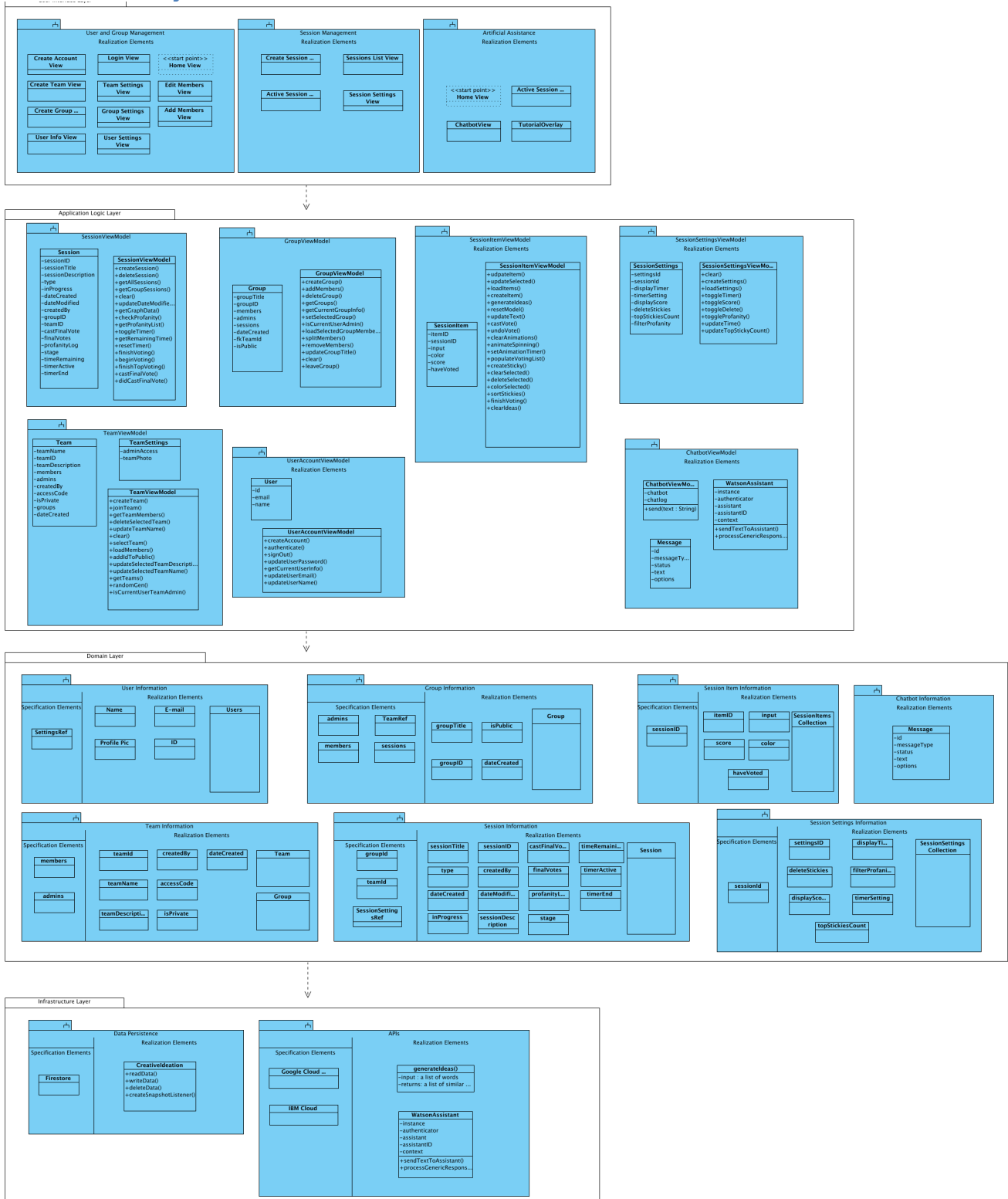
4.3.3 Structural Realization - Manage Teams.....	21
4.4 Manage Sessions.....	21
4.4.1 Behavioural Model - Manage Sessions.....	21
4.4.2 Collaboration Overview - Manage Sessions	22
4.4.3 Structural Realization - Manage Sessions.....	22
4.5 Collaborate with Peers.....	22
4.5.1 Behavioural Model - Collaborate with Peers.....	23
4.5.2 Collaboration Overview - Collaborate with Peers	23
4.5.3 Structural Realization - Collaborate with Peers.....	24
4.6 Guide Session.....	24
4.6.1 Behavioural Model - Guide Session.....	24
4.6.2 Collaboration Overview - Guide Session	25
4.6.3 Structural Realization - Guide Session.....	25
4.7 Generate Ideas.....	25
4.7.1 Behavioural Model - Generate Ideas.....	26
4.7.2 Collaboration Overview - Generate Ideas	26
4.7.3 Structural Realization - Generate Ideas.....	27
4.8 Get Help	27
4.8.1 Behavioural Model - Get Help	27
4.8.2 Collaboration Overview - Get Help.....	28
4.8.3 Structural Realization - Get Help	28

1. Introduction

This document provides a detailed overview of the Creative Ideation application's software architecture design. This document serves as a guide through the architectural layout of our application; providing a description of the software system architecture, the overview diagrams of each architectural layer, and the interaction model diagrams for each major use case.

2. Software Architecture

2.1. Software System Architecture



Creative Ideation is an Event Processing System – more specifically, an Editing System. The Editing System reference architecture describes a system that is meant for collaboration between users, provides rapid feedback to user actions, is responsive and easy to use. The fundamental structural organization schema is expressed by a layered architecture pattern, as our system appropriately fits this pattern. Creative Ideation benefits from a layered pattern because the app is

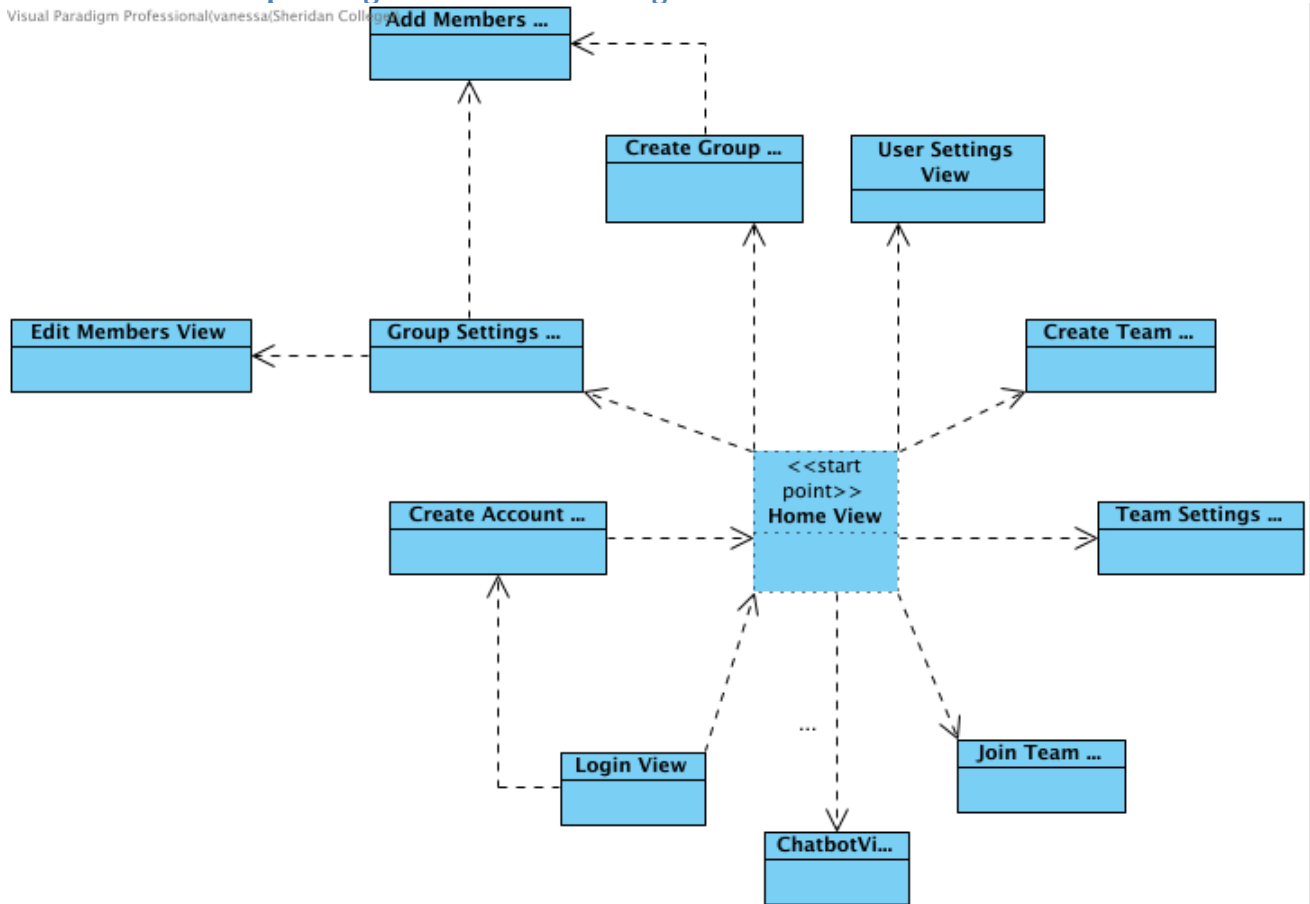
designed to work on multiple screen types/devices, and the simplicity of the pattern is complimented by the Model–View–View Model (MVVM) structural design pattern.

3. Detailed Design

3.1. User Interface Layer

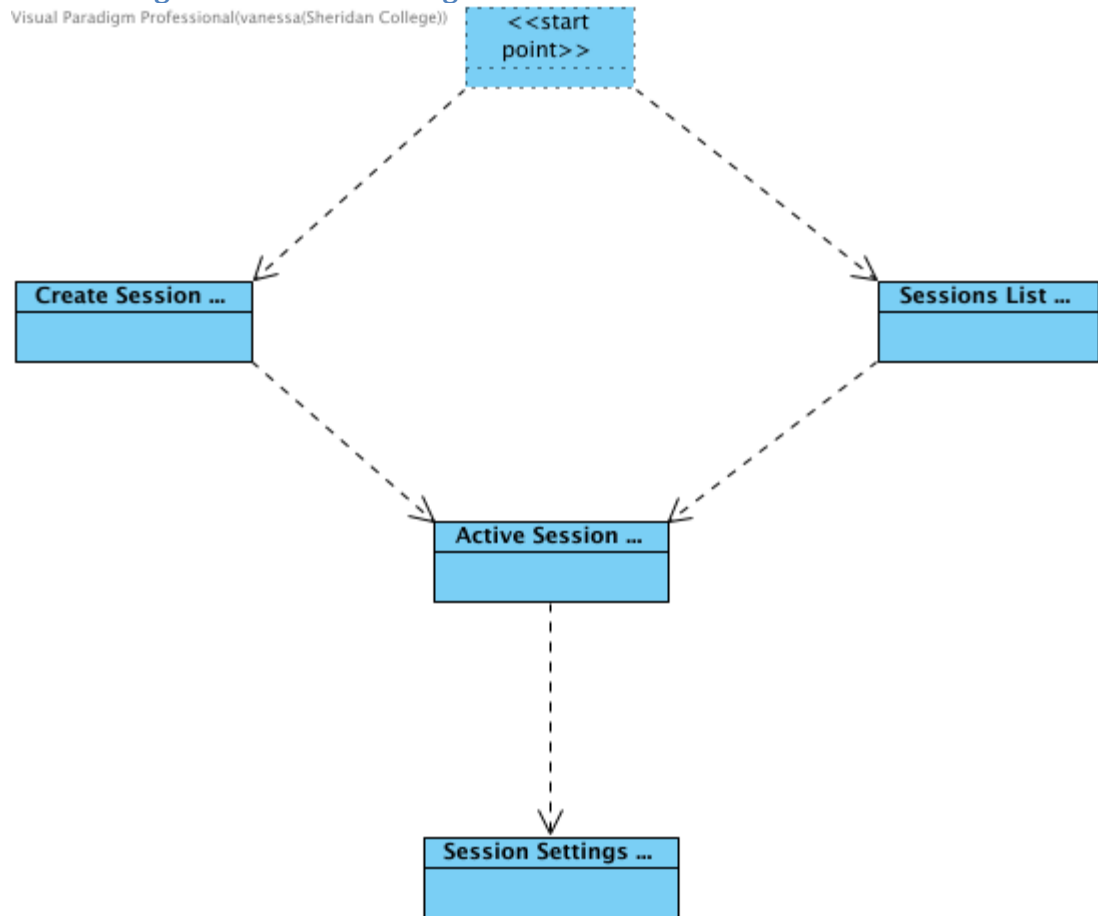
This layer describes all interfaces that users may interact with within the application. All subsystems correspond to UI elements found within the iPad application. This layer is divided into three subsystems each describing a distinct functional area: User and Group Management, Session Management, and Artificial Assistance.

3.1.1. User and Group Management Overview Diagram



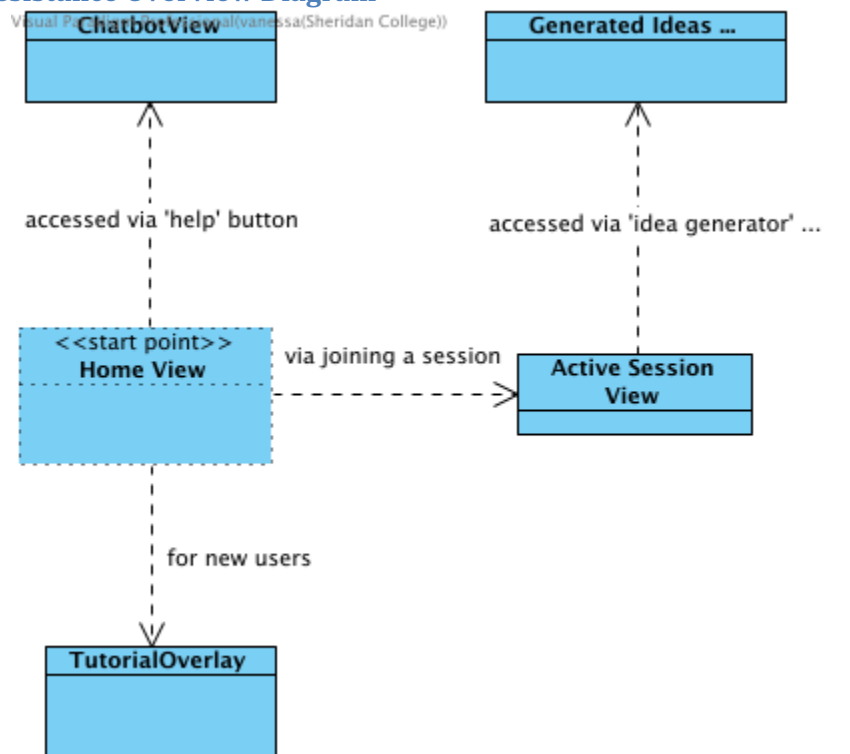
This model shows how each view in this functional area interacts.

3.1.2. Session Management Overview Diagram



This model shows the interaction between views in the Session Management functional area.

3.1.3. Artificial Assistance Overview Diagram

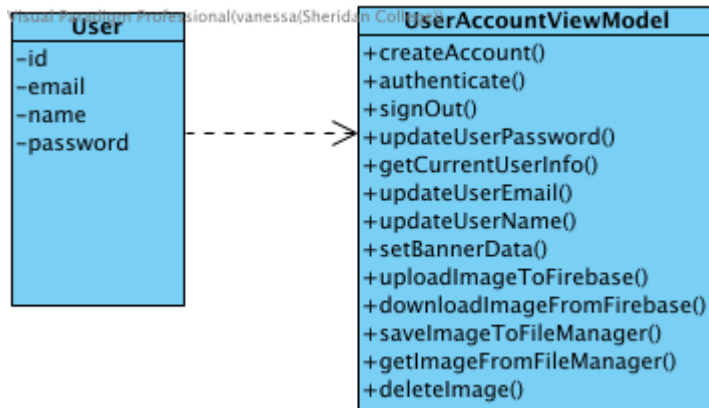


This model shows the interaction between views in the Artificial Assistance functional area.

3.2. Application Logic Layer

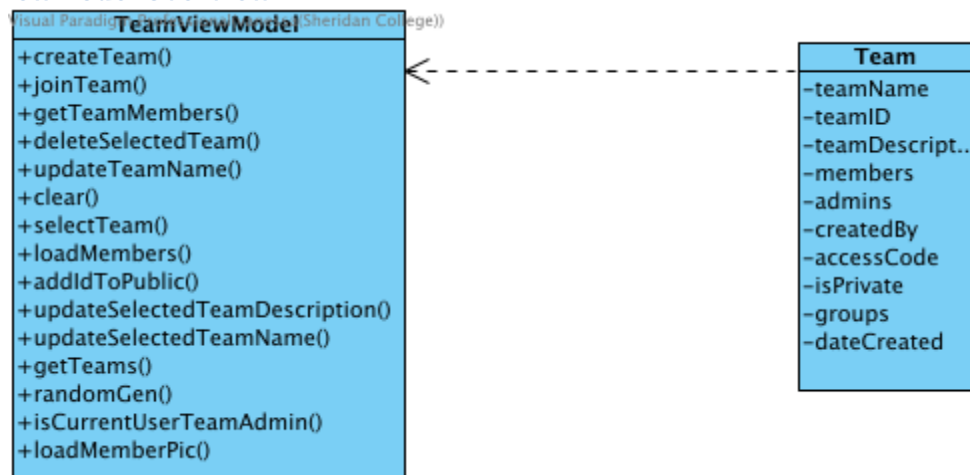
The Application Logic Layer describes all models/view models that handle business logic within the application. Each subsystem within this layer details operations that handle data and dictate what the UI layer has access to. The subsystems of the UI layer directly depend on the data housed in this layer due to the declarative nature of SwiftUI.

3.2.1. UserViewModel Overview



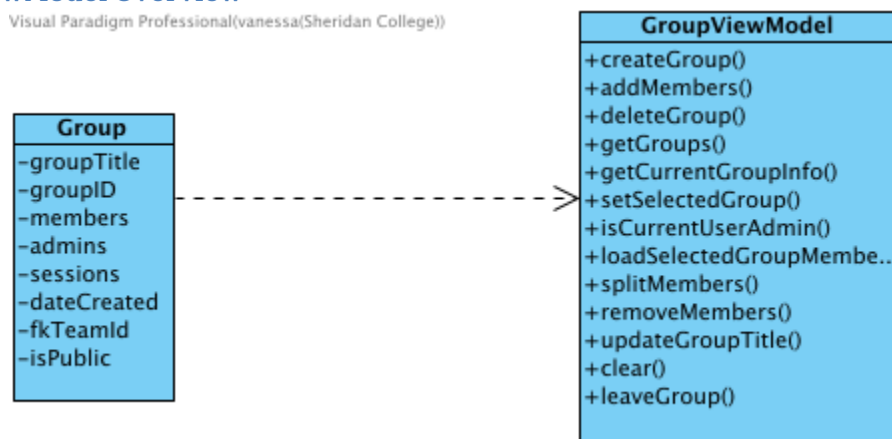
This model shows how the UserAccountViewModel interacts with User and UserSettings model objects.

3.2.2. TeamViewModel Overview



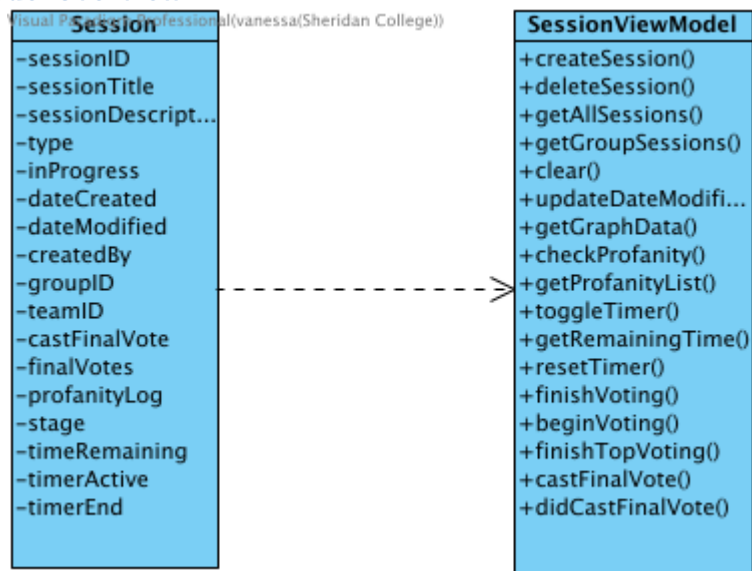
This model shows how the TeamViewModel interacts with Team and TeamSettings model objects.

3.2.3. GroupViewModel Overview



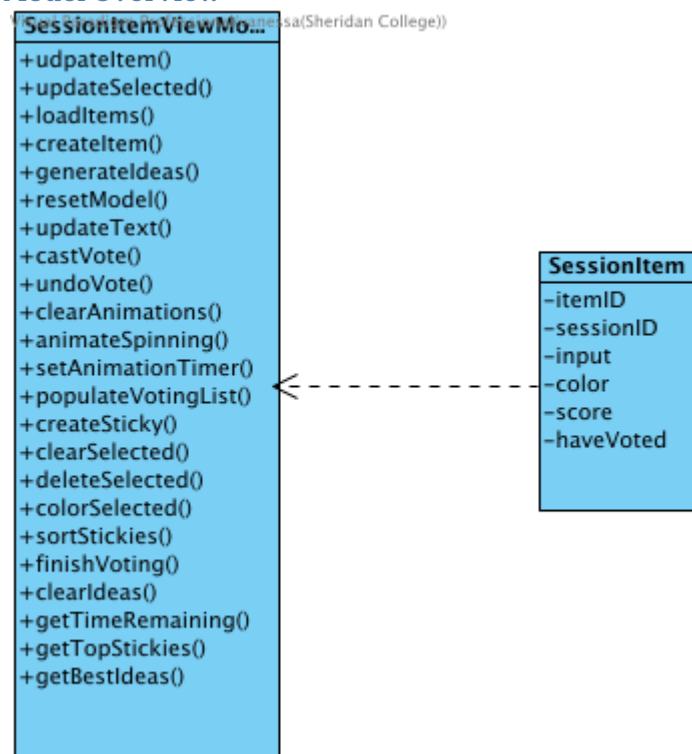
This model shows how the GroupViewModel interacts with Group model objects.

3.2.4. SessionViewModel Overview



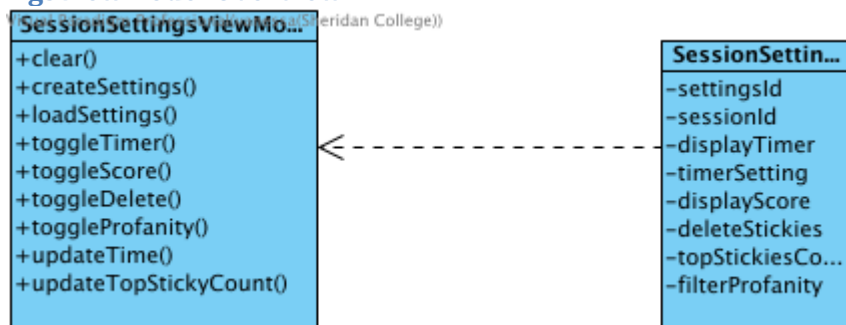
This model shows how the SessionViewModel interacts with Session model objects.

3.2.5. SessionItemViewModel Overview



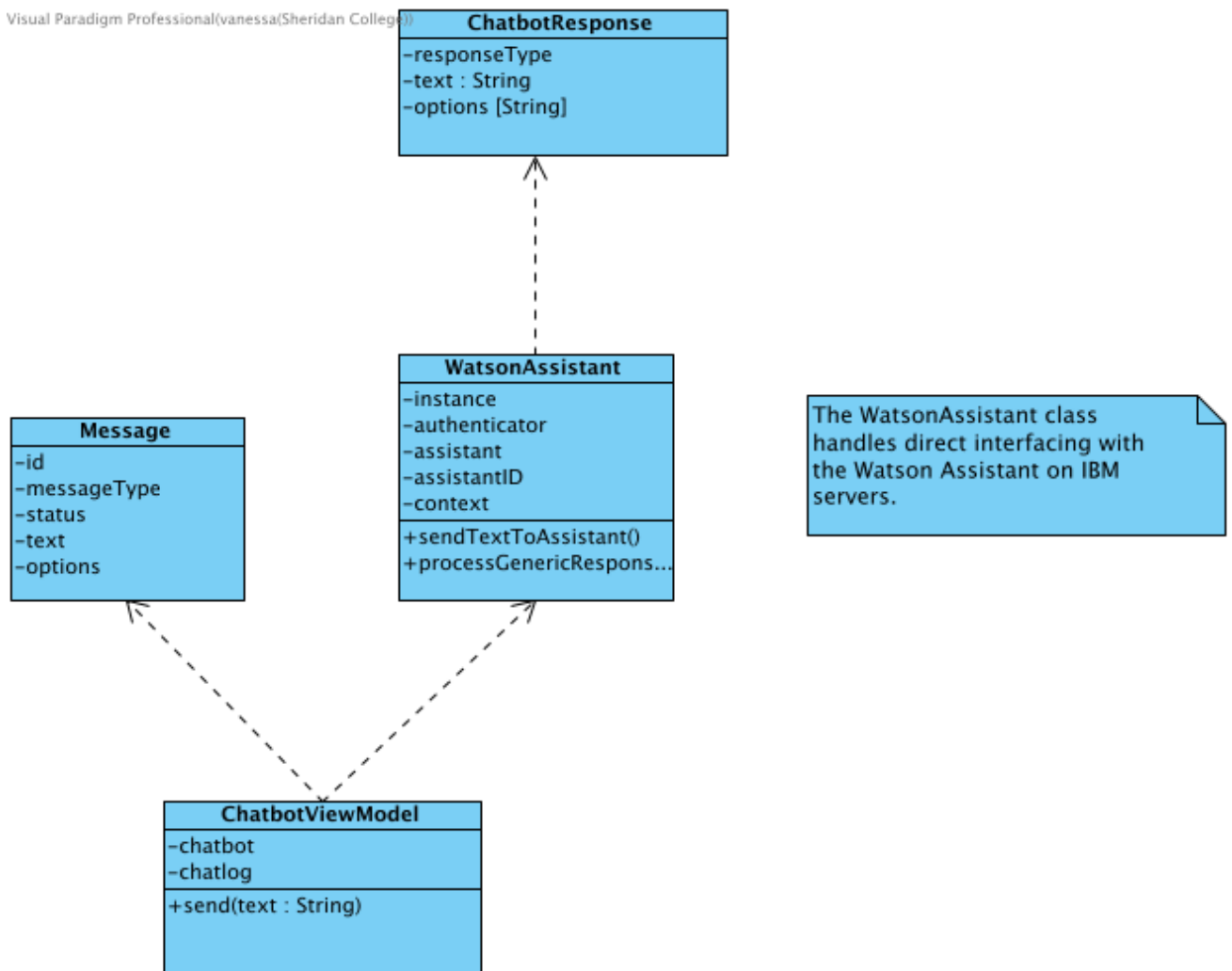
This model shows how the SessionItemViewModel interacts with SessionItem model objects.

3.2.6. SessionSettingsViewModel Overview



This model shows how the SessionSettingsViewModel interacts with SessionSettings model objects.

3.2.7. ChatbotViewModel Overview



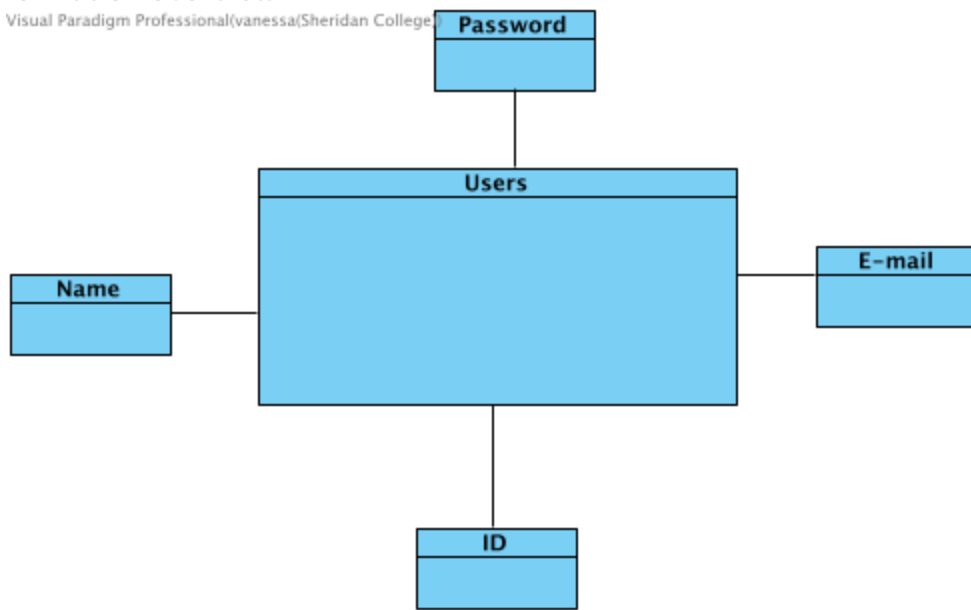
This model shows how the ChatbotViewModel interacts with Message model objects, the WatsonAssistant interface, and the ChatbotView.

3.3. Domain Layer

This layer describes conceptual models used by the Application Layer to represent data in structural format. Some of the models detailed in this layer correspond to data structures found outside of the application (remote database). For example, the User information model can be used as an interface between the Firestore database (containing User information) and the UserViewModel to allow for seamless communication between the database and mobile app.

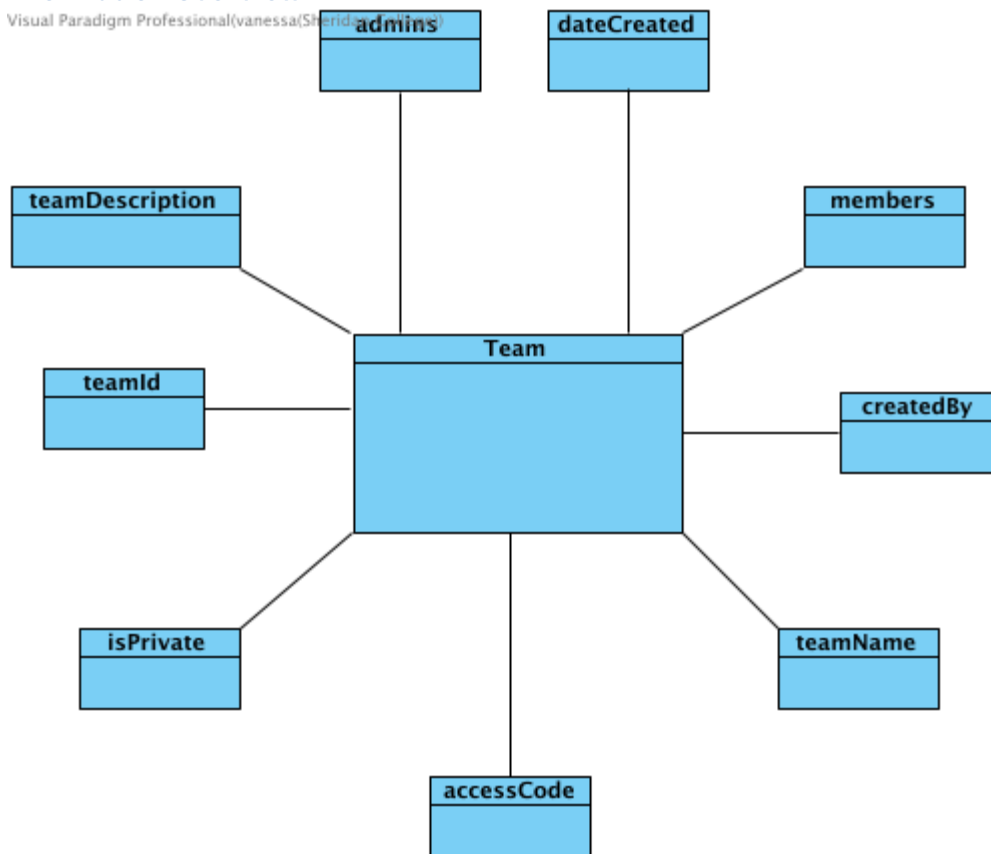
3.3.1. User Information Overview

Visual Paradigm Professional(vanessa(Sheridan College)



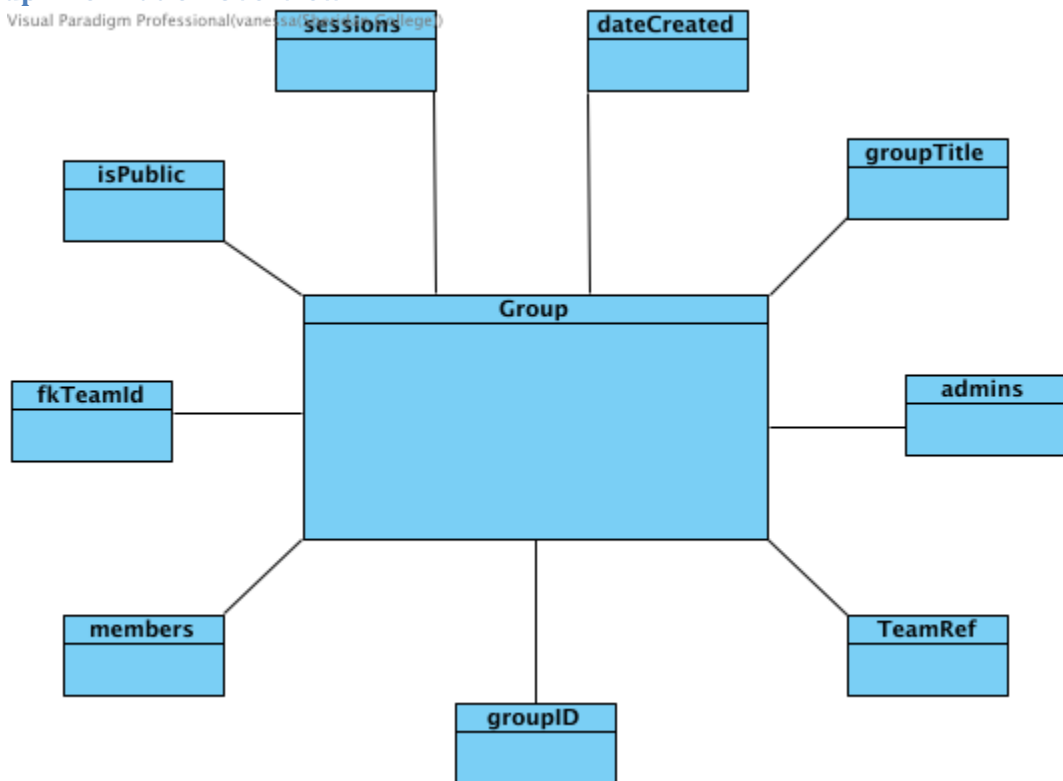
3.3.2. Team Information Overview

Visual Paradigm Professional(vanessa(Sheridan College)



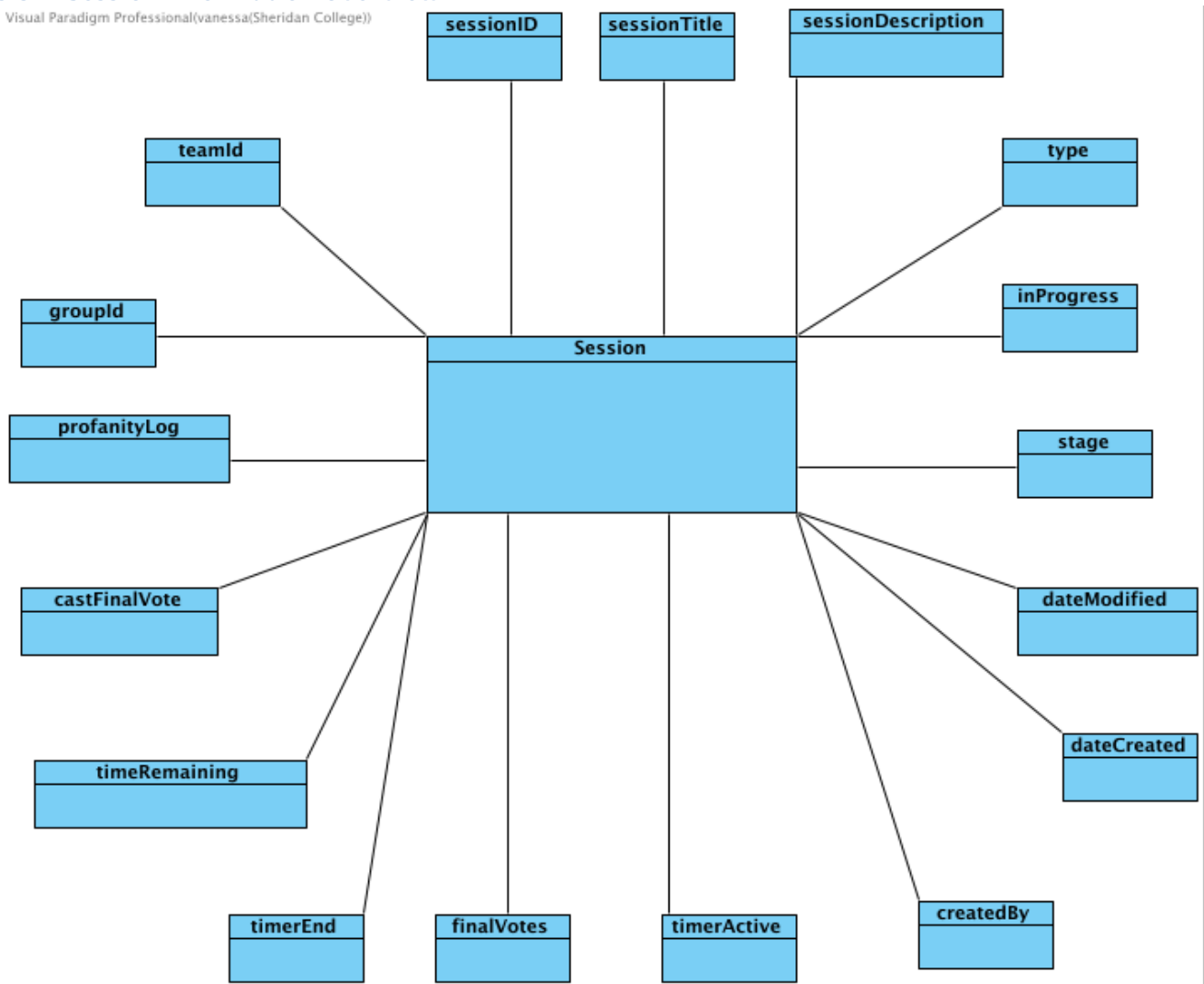
3.3.3. Group Information Overview

Visual Paradigm Professional(vanessa@knightcollege.edu)



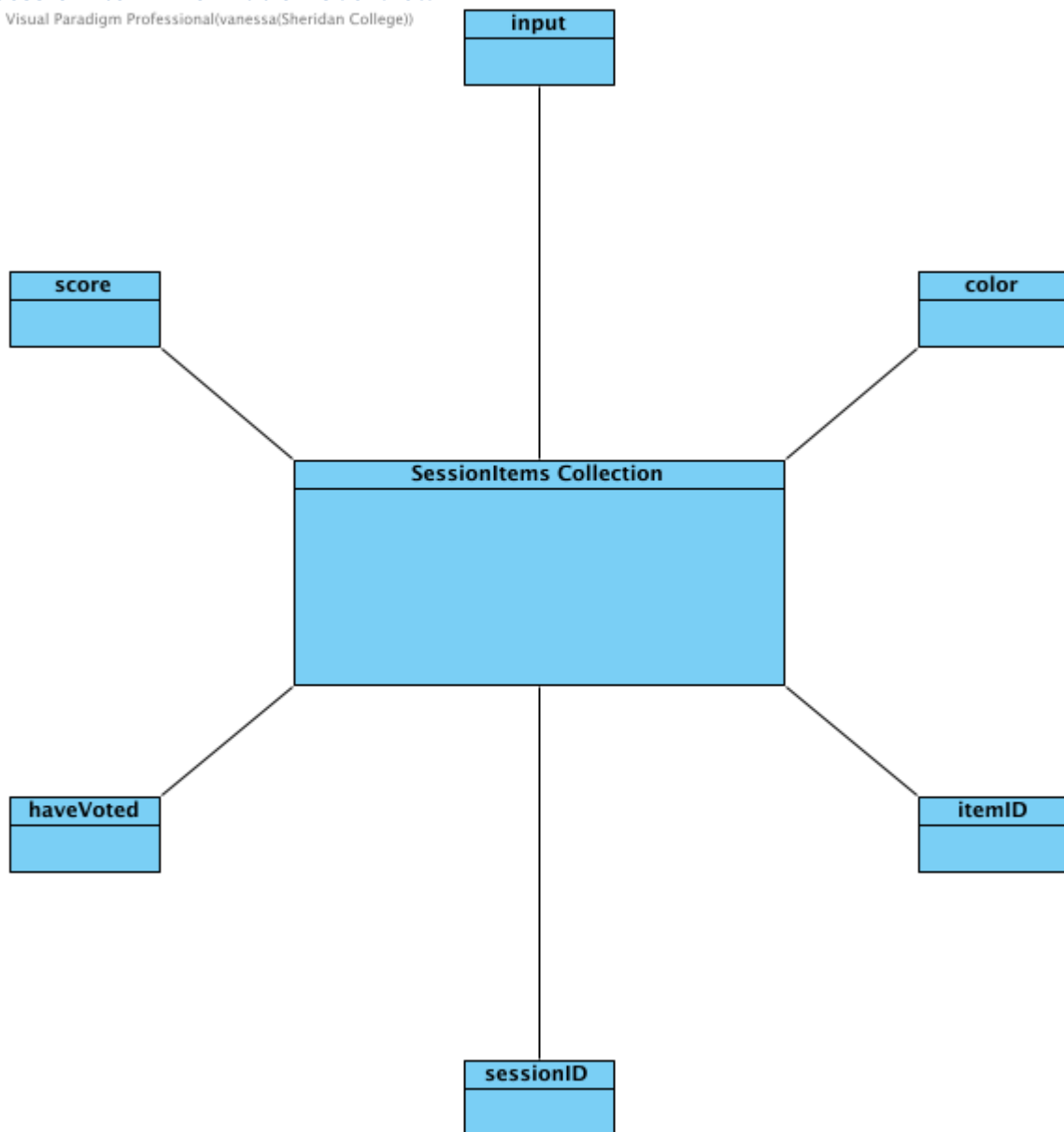
3.3.4. Session Information Overview

Visual Paradigm Professional(vanessa(Sheridan College))



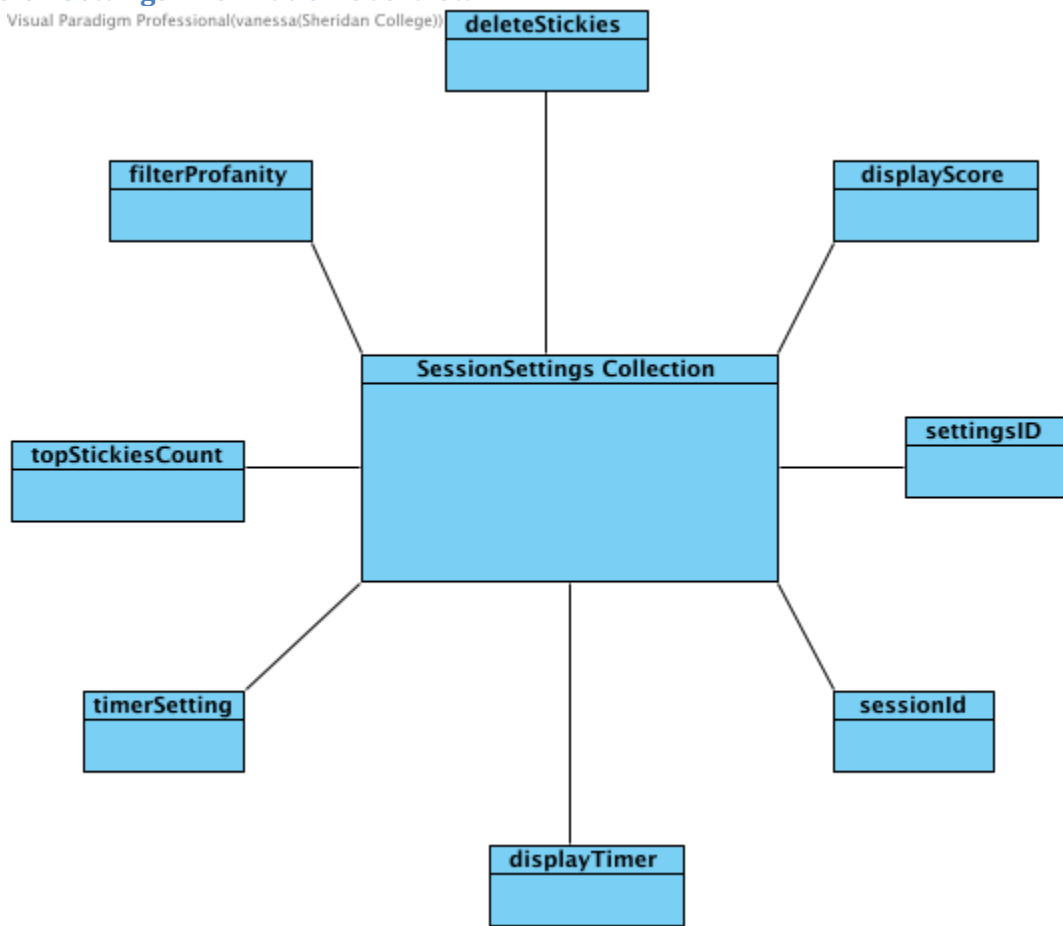
3.3.5. Session Item Information Overview

Visual Paradigm Professional(vanessa(Sheridan College))



3.3.6. Session Settings Information Overview

Visual Paradigm Professional(vanessa(Sheridan College))

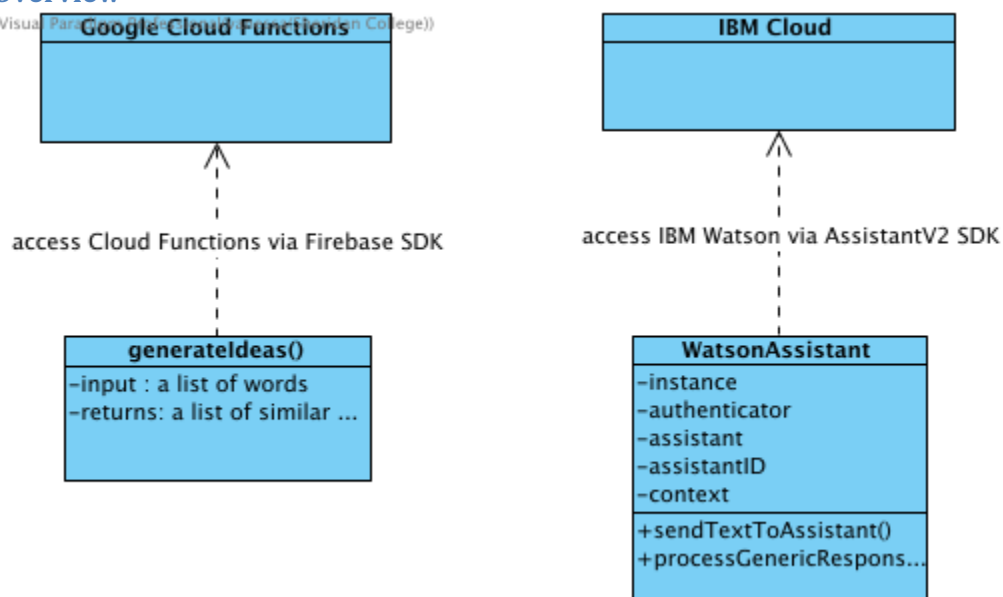


3.4. Infrastructure Layer

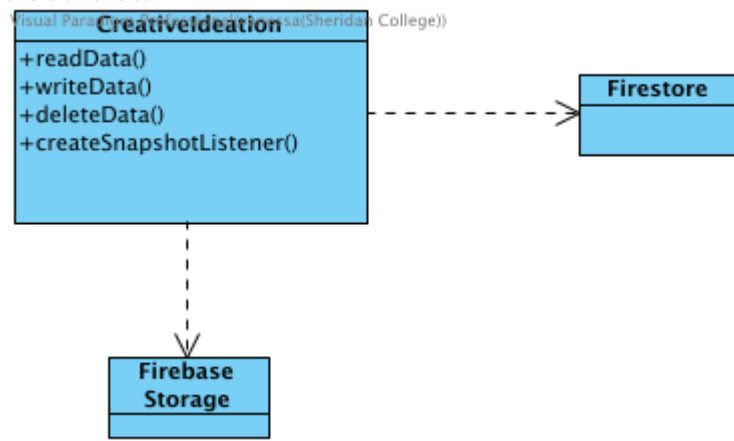
This layer describes data storage infrastructure, as well as APIs the application is dependant on.

3.4.1. APIs Overview

Visual Paradigm Professional(vanessa(Sheridan College))



3.4.2. Data Persistence Overview

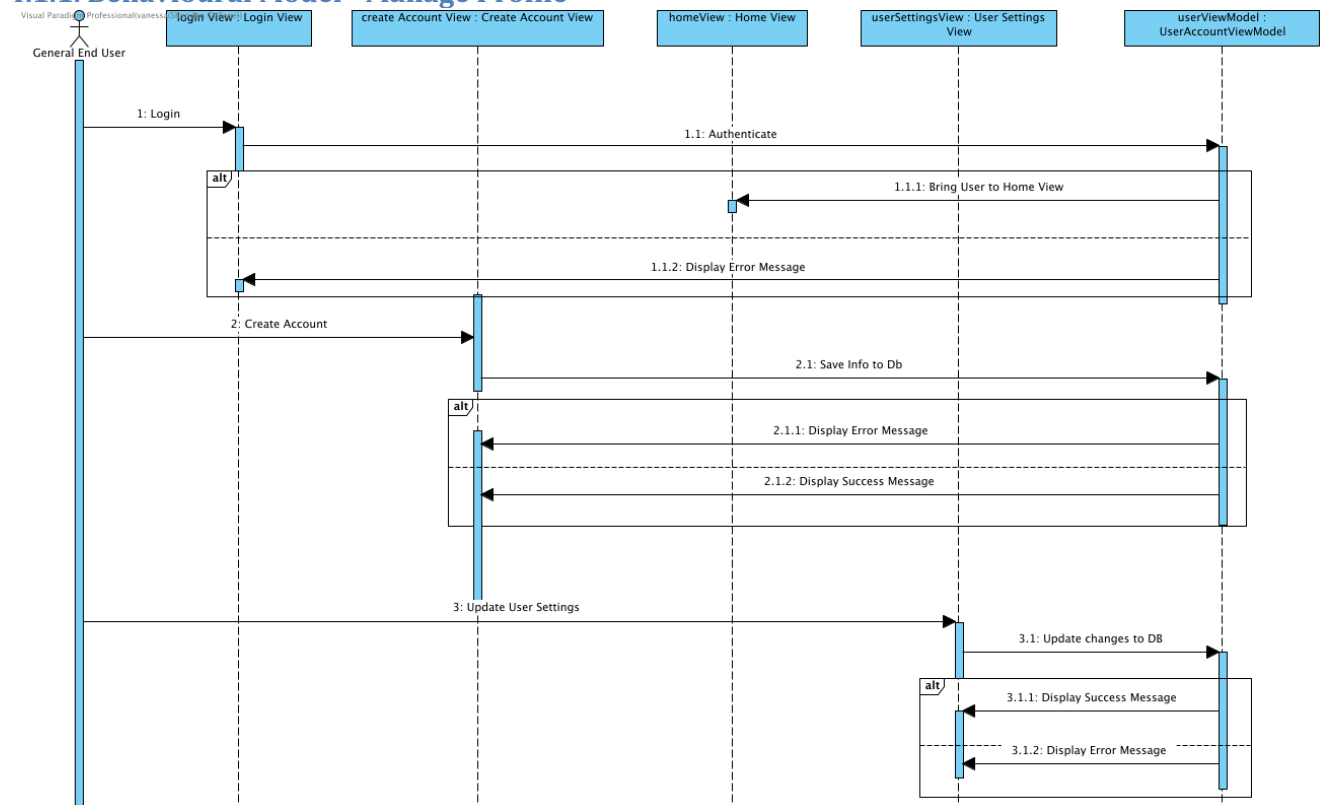


4. Interaction Model

4.1. Manage Profile

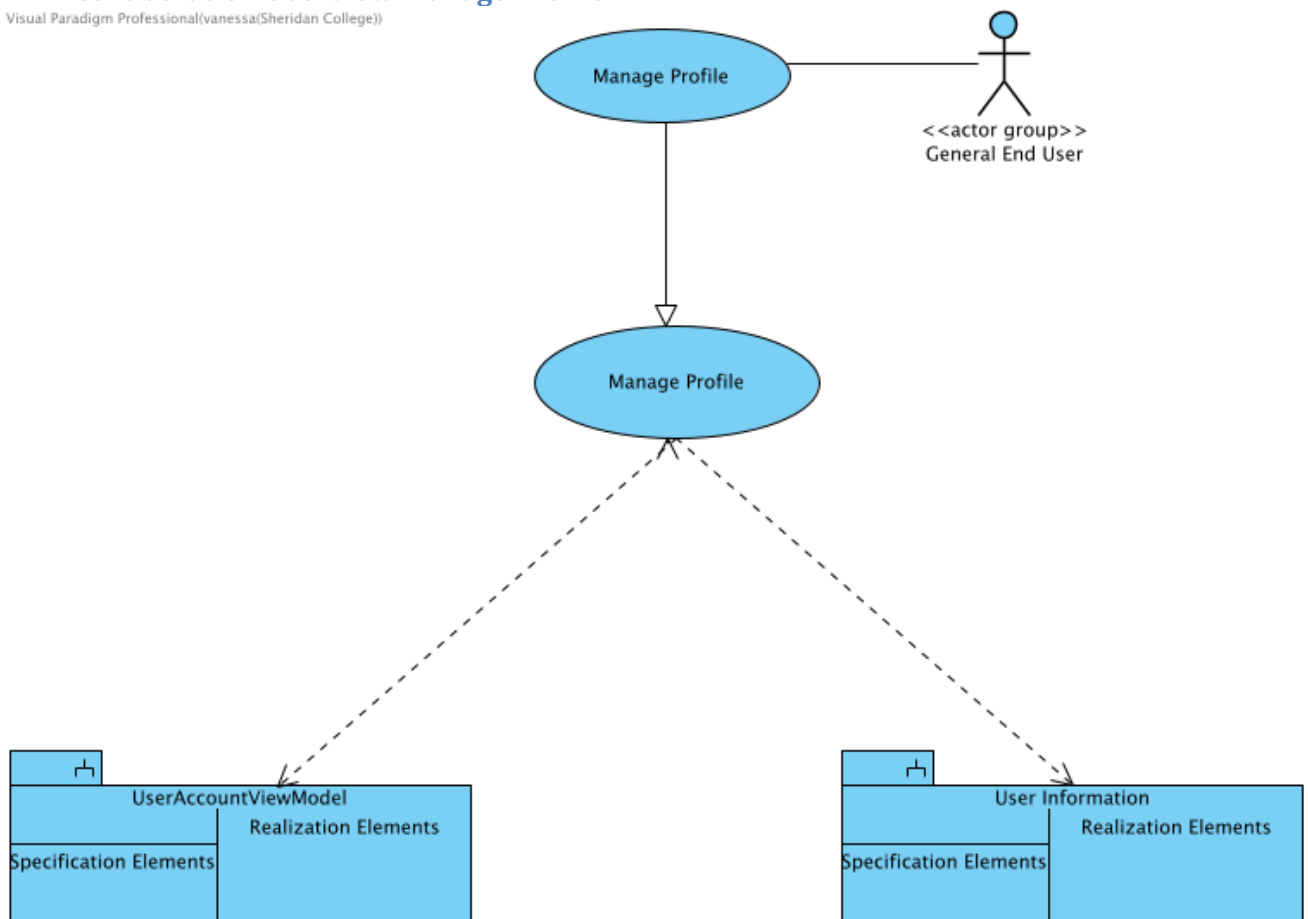
Users will be able to create their profile, update their credentials, manage their settings, and log out of their account

4.1.1. Behavioural Model - Manage Profile



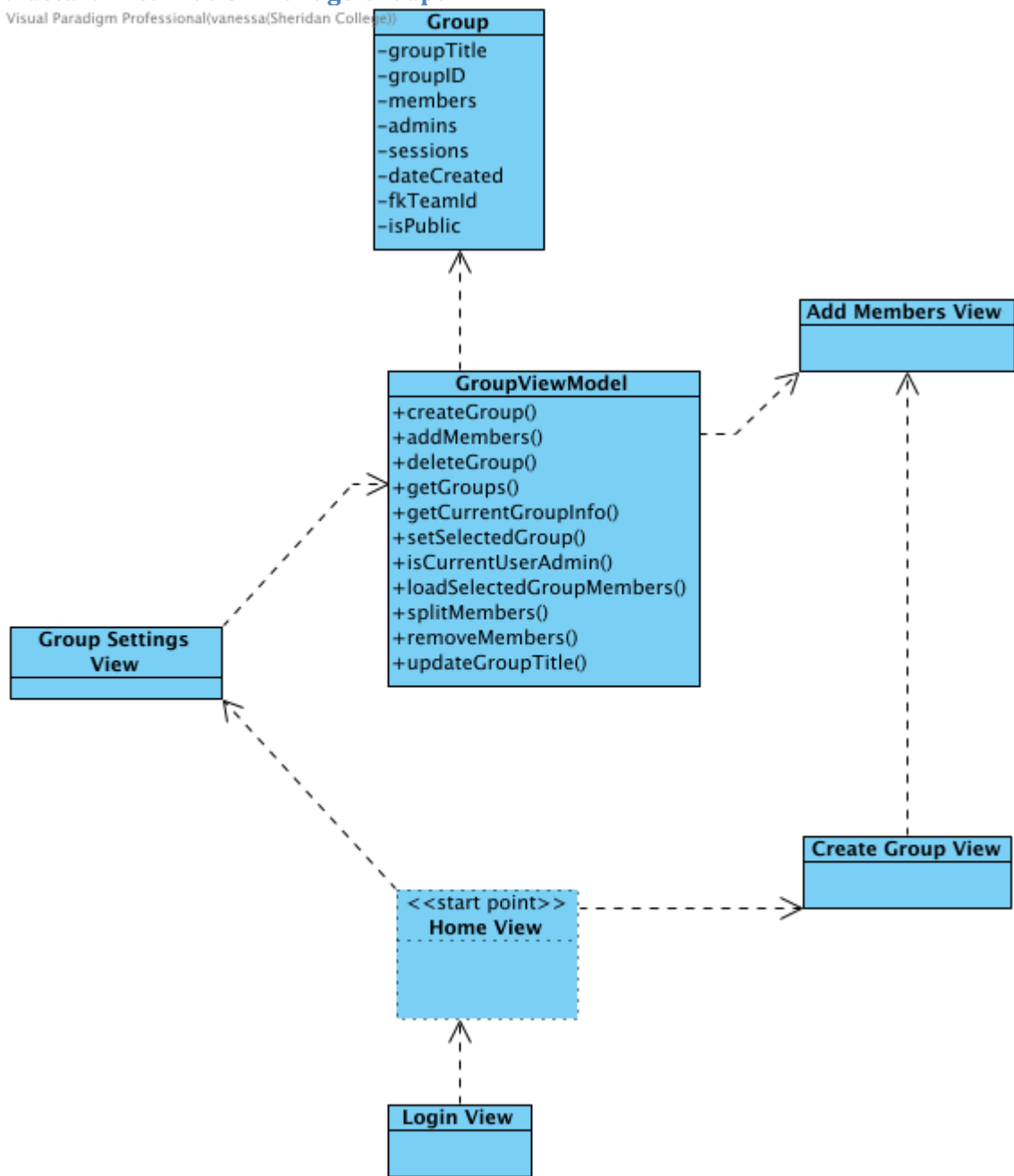
4.1.2. Collaboration Overview-Manage Profile

Visual Paradigm Professional(vanessa(Sheridan College))



4.1.3. Structural Realization Manage Groups

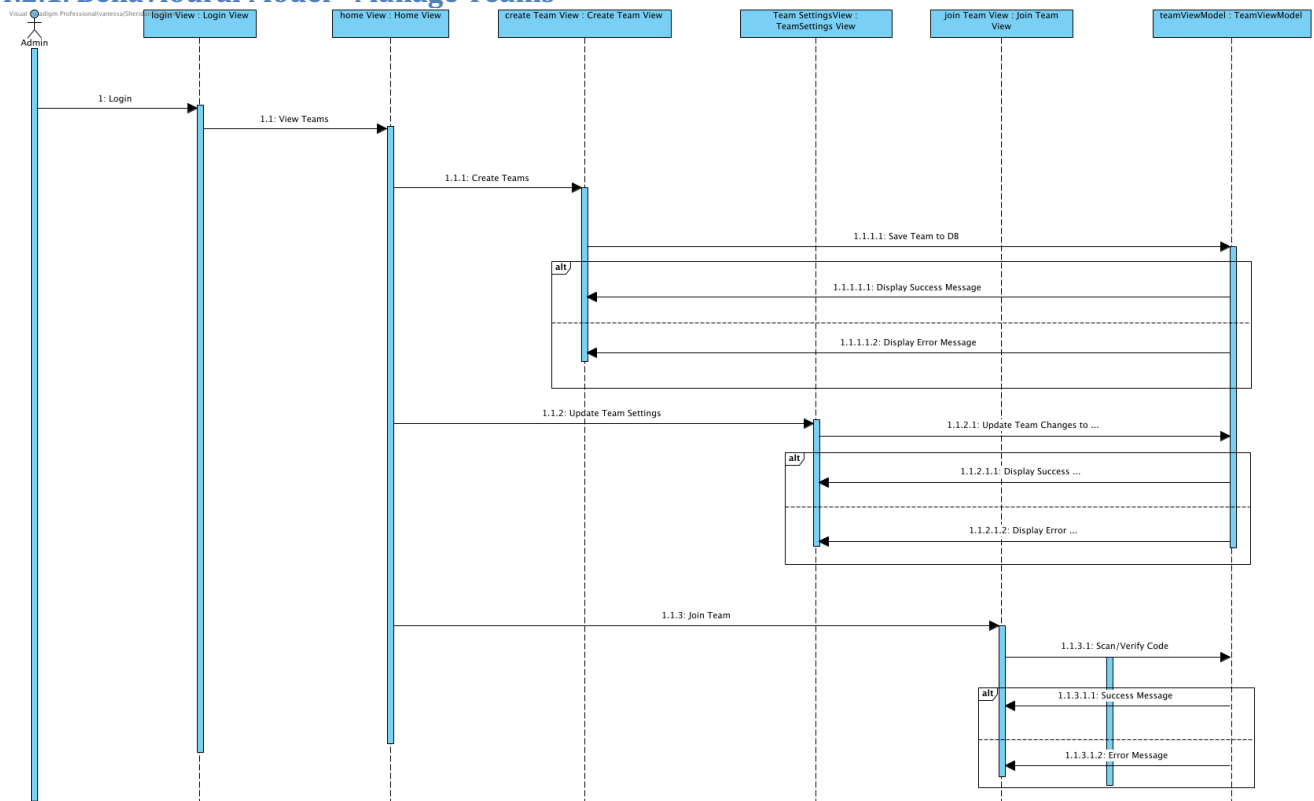
Visual Paradigm Professional(vanessa(Sheridan College))



4.2. Manage Teams

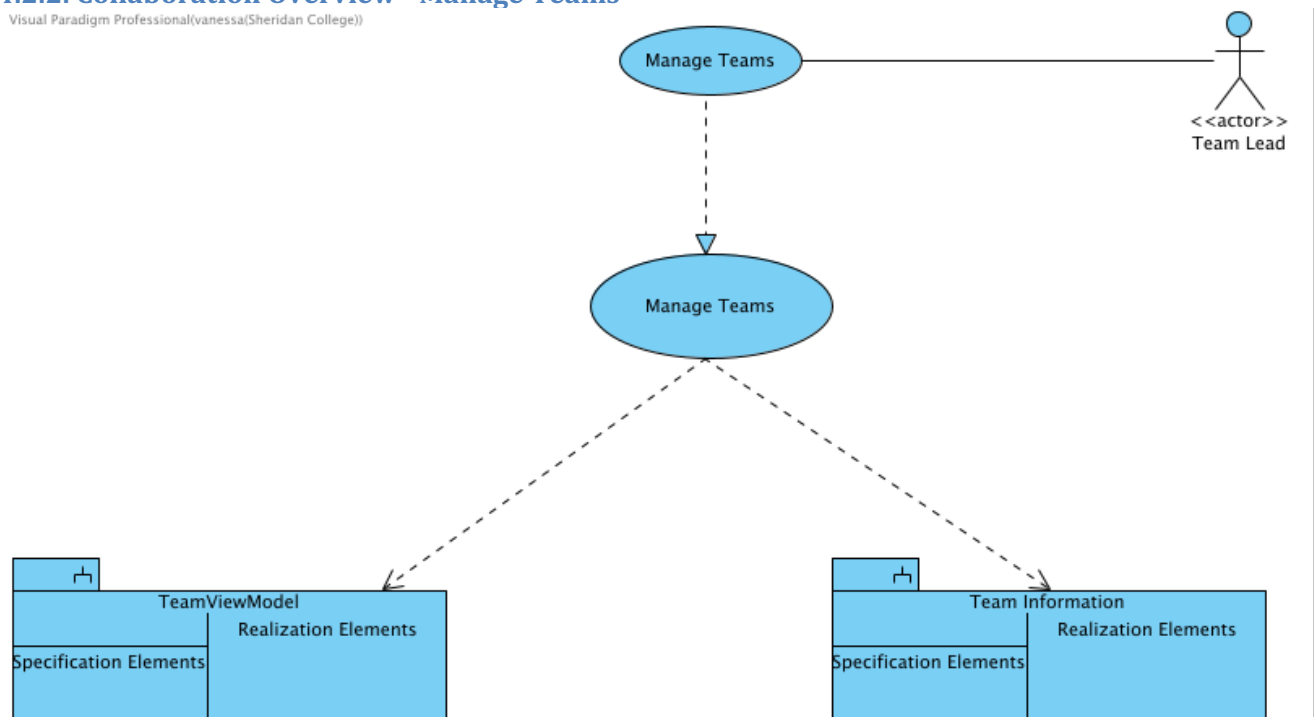
Users will be able to create, add people to, edit settings for, and delete teams.

4.2.1. Behavioural Model - Manage Teams



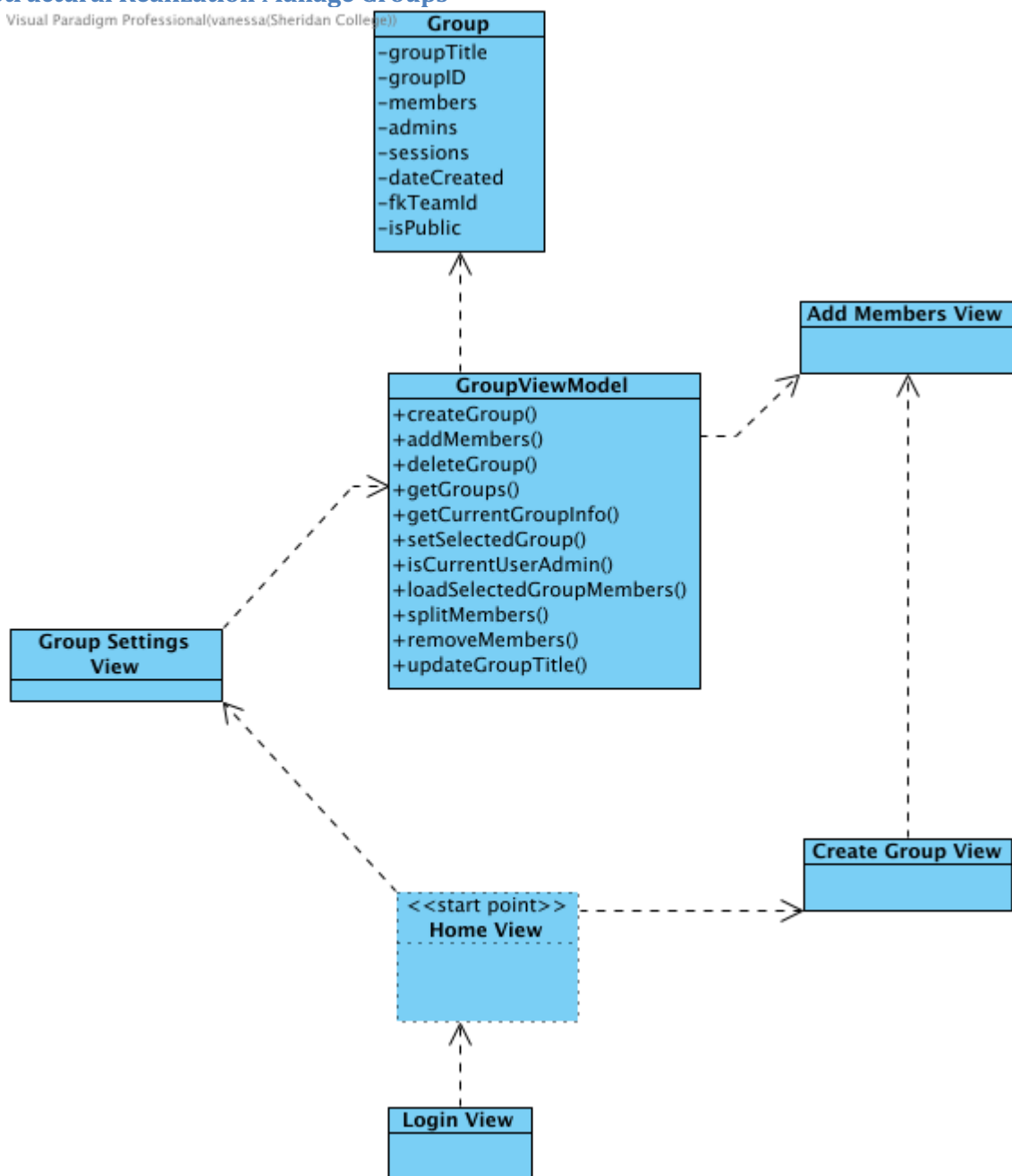
4.2.2. Collaboration Overview - Manage Teams

Visual Paradigm Professional(vanessa(Sheridan College))



4.2.3. Structural Realization Manage Groups

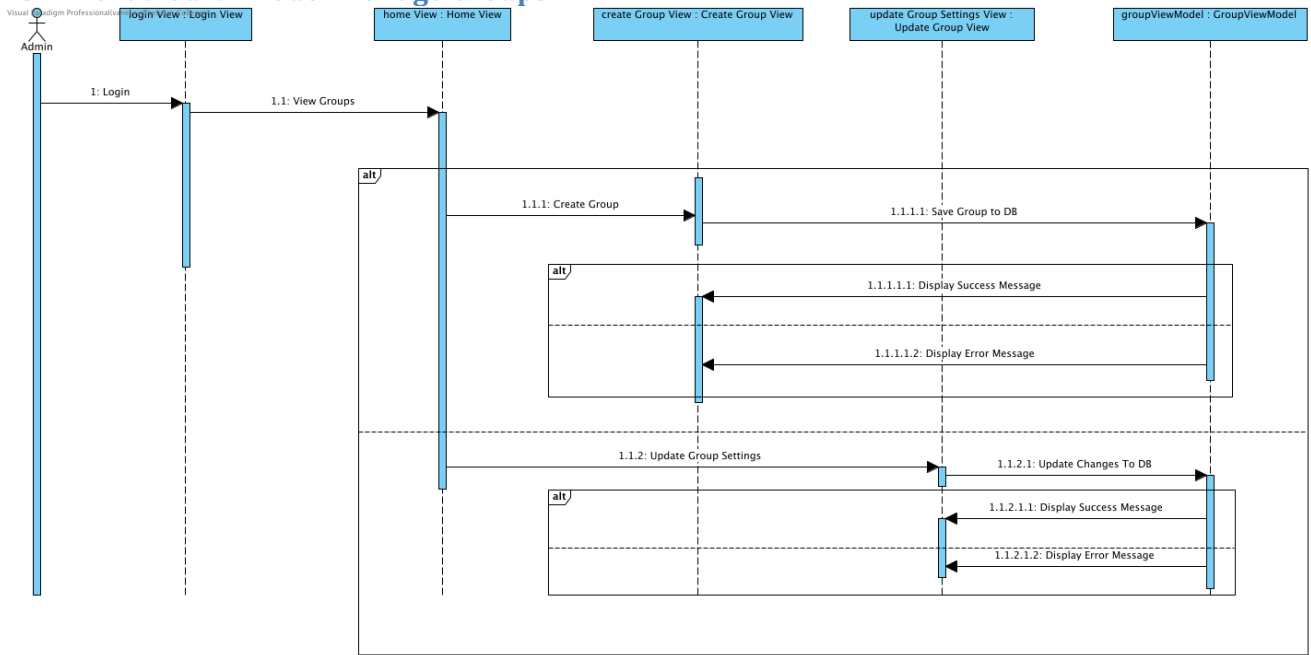
Visual Paradigm Professional(vanessa(Sheridan College))



4.3. Manage Groups

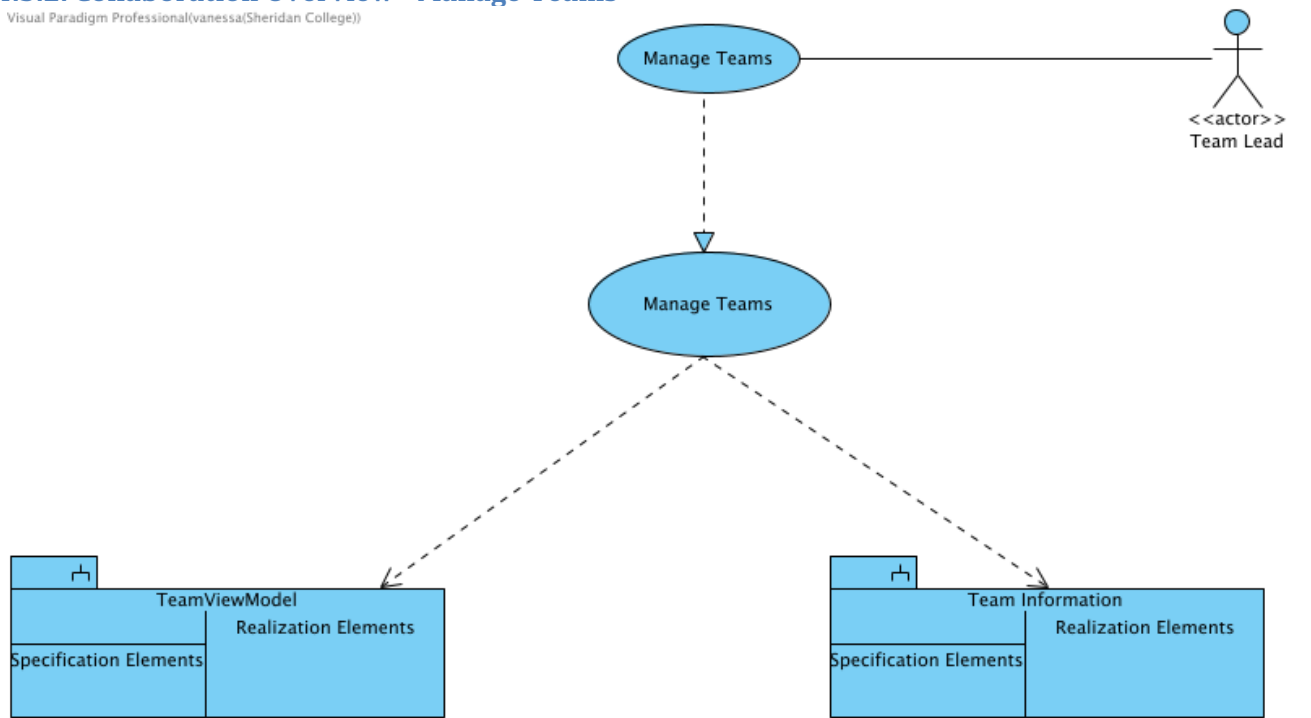
Users will be able to create, add people to, change settings for, and delete groups within a team.

4.3.1. Behavioural Model-Manage Groups

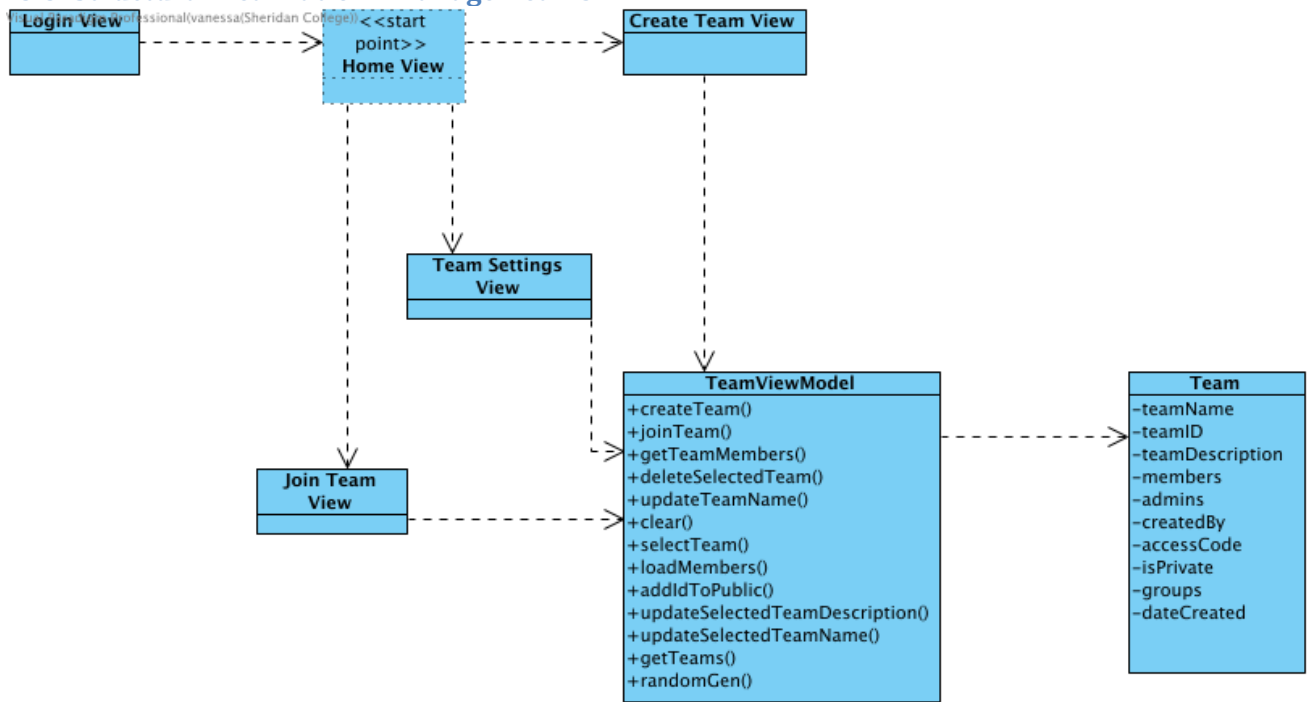


4.3.2. Collaboration Overview - Manage Teams

Visual Paradigm Professional(vanessa(Sheridan College))



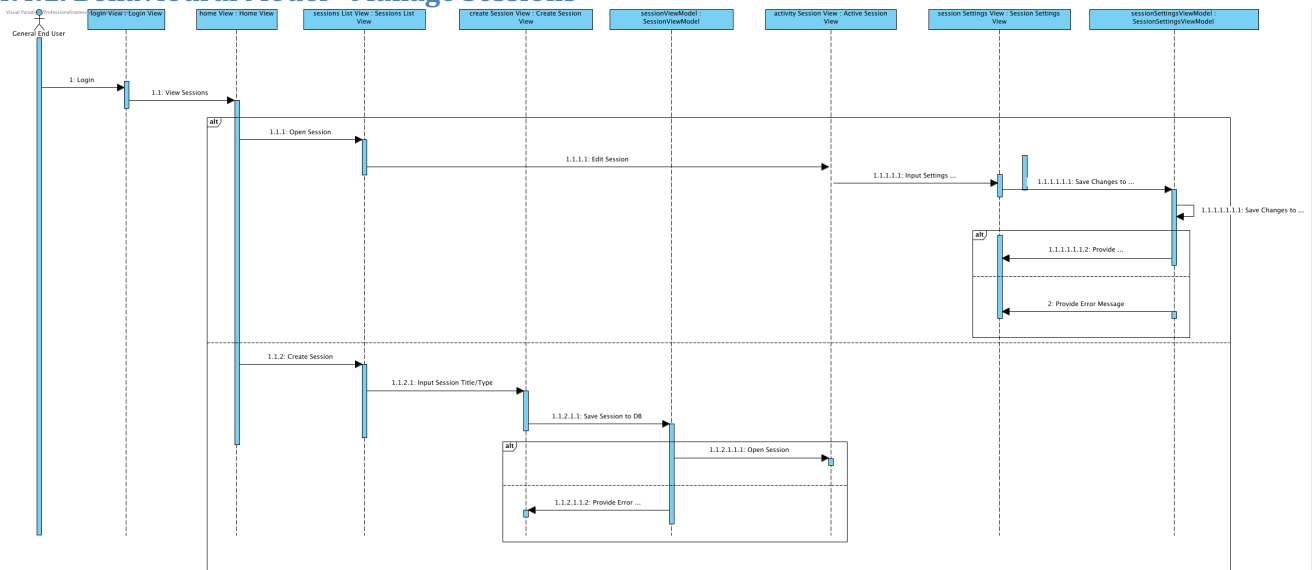
4.3.3. Structural Realization - Manage Teams



4.4. Manage Sessions

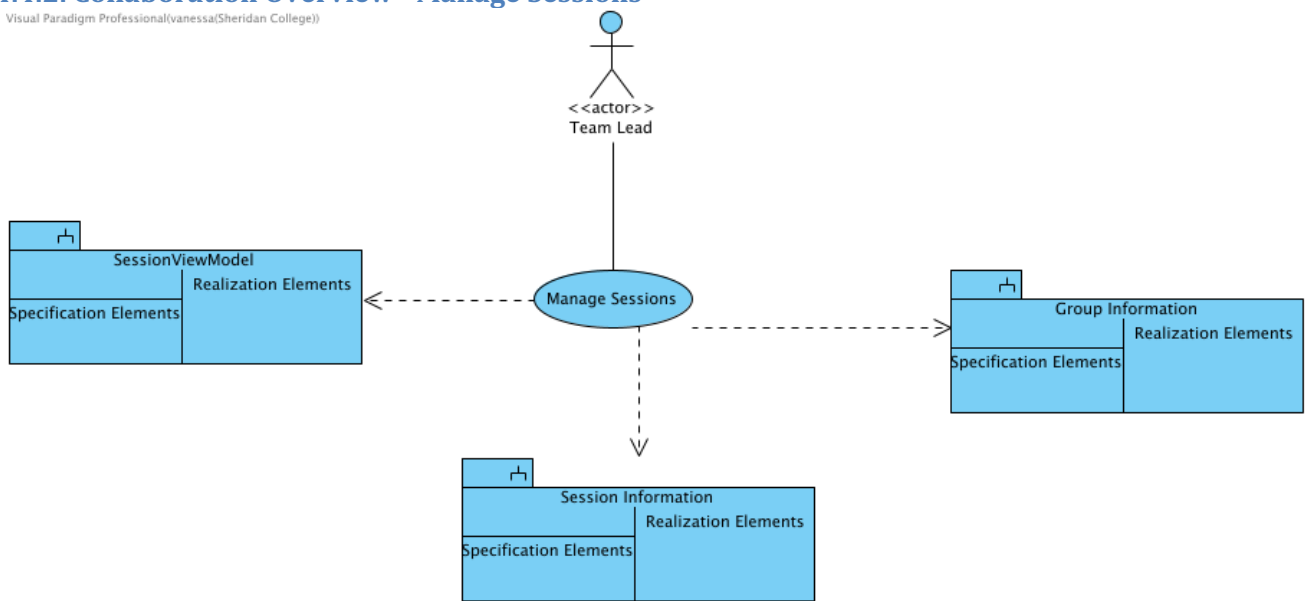
Users will be able to create sessions, and each session will have a settings page that allows the session creator to choose settings, such as profanity filtering

4.4.1. Behavioural Model - Manage Sessions



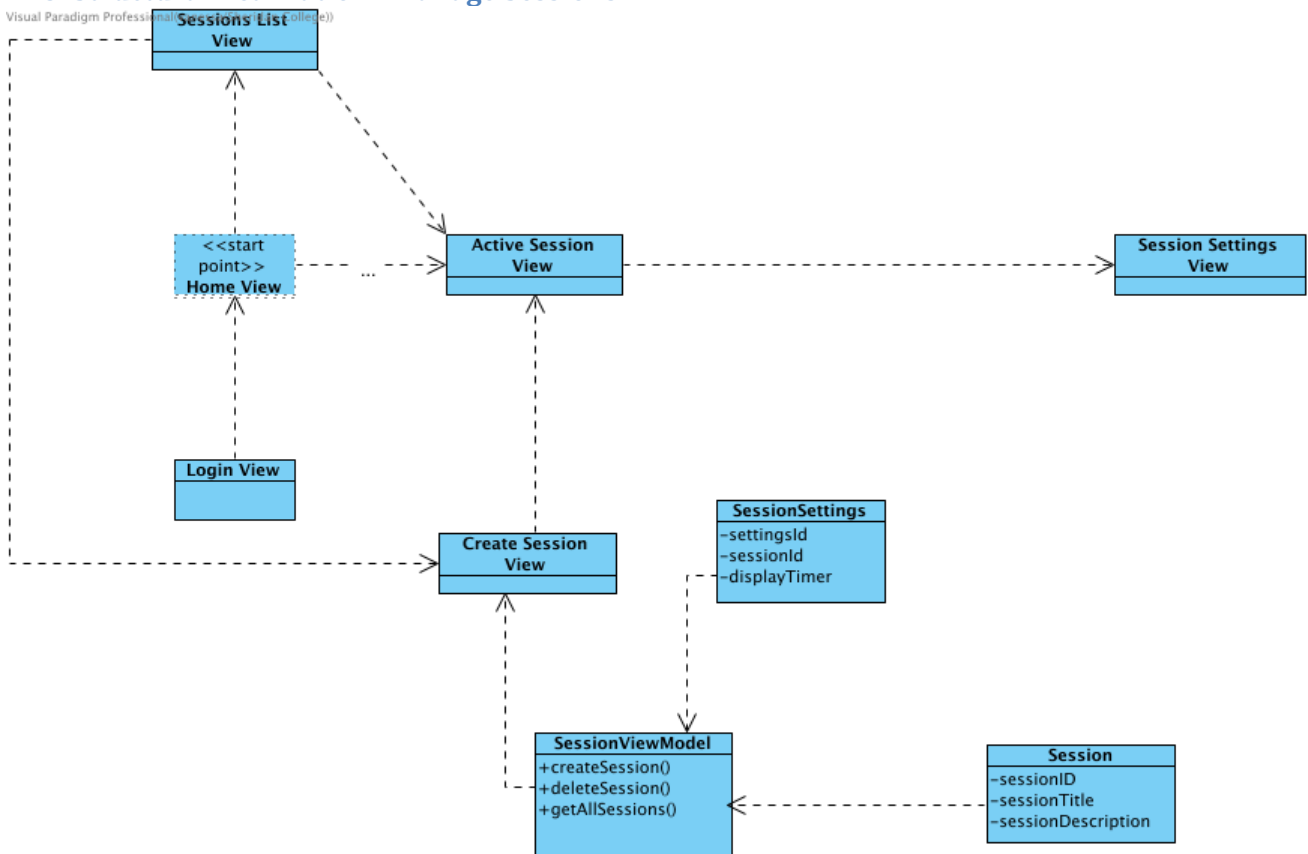
4.4.2. Collaboration Overview - Manage Sessions

Visual Paradigm Professional(vanessa(Sheridan College))



4.4.3. Structural Realization - Manage Sessions

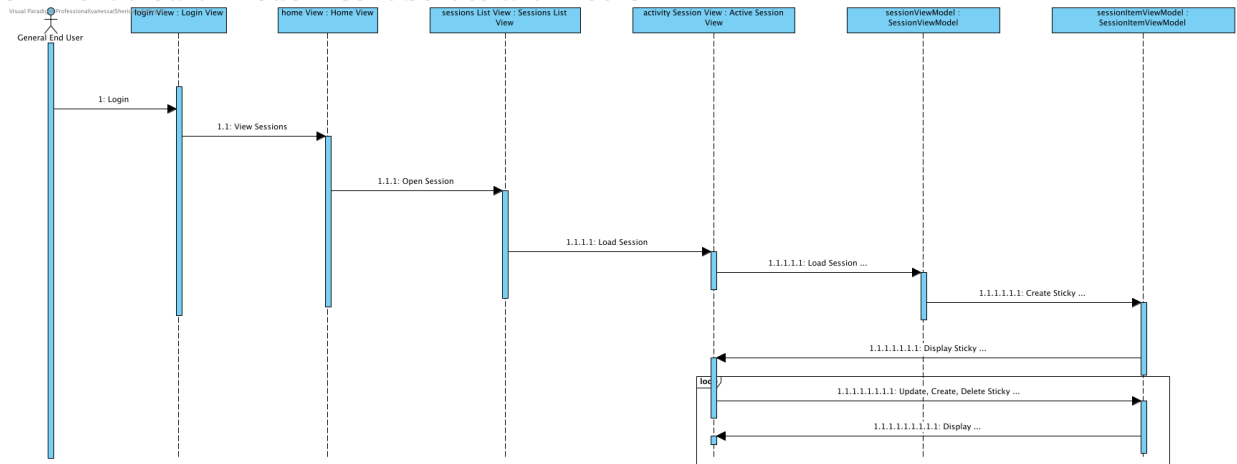
Visual Paradigm Professional(vanessa(Sheridan College))



4.5. Collaborate with Peers

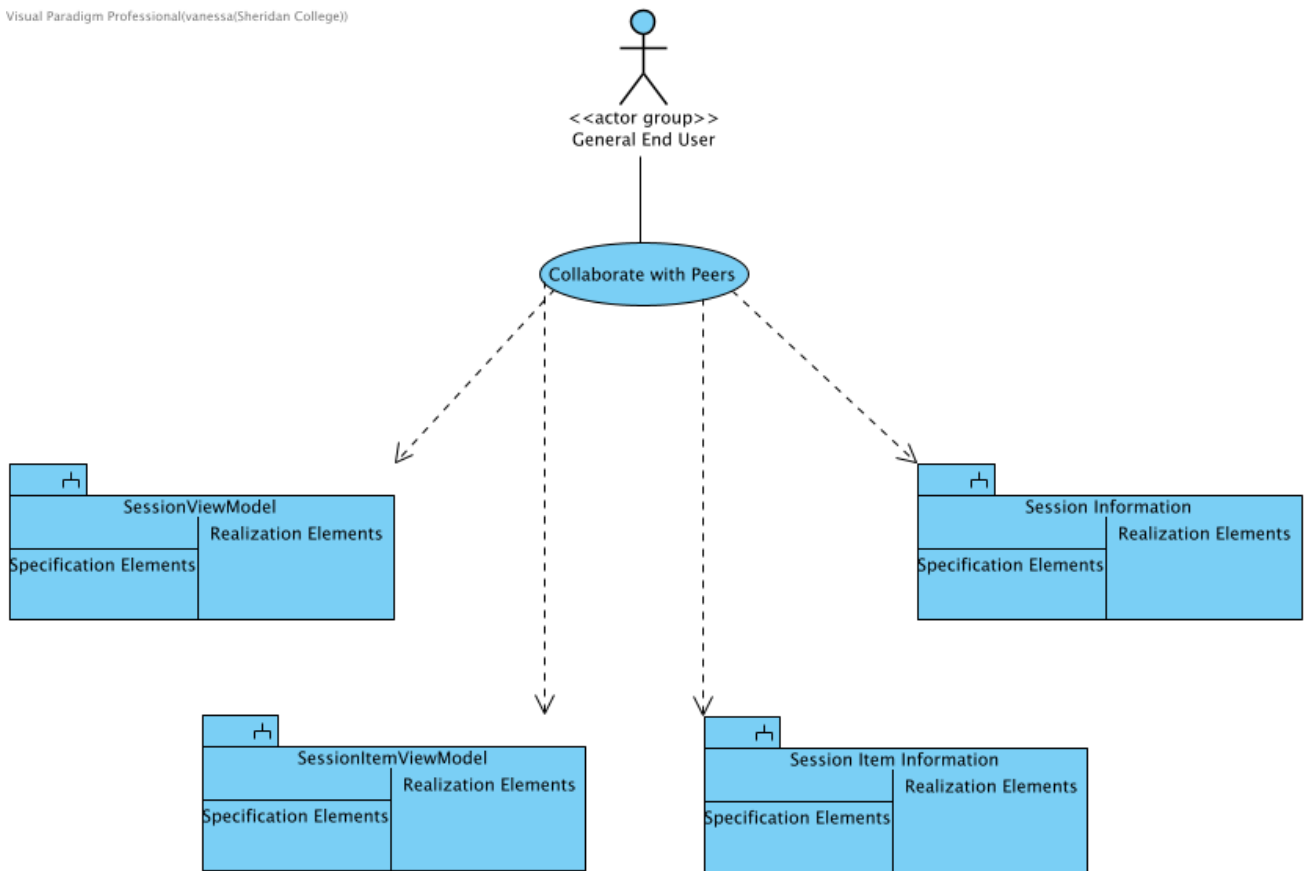
Users will be able to collaborate with peers in real time so that they can brainstorm/participate in sessions together

4.5.1. Behavioural Model - Collaborate with Peers

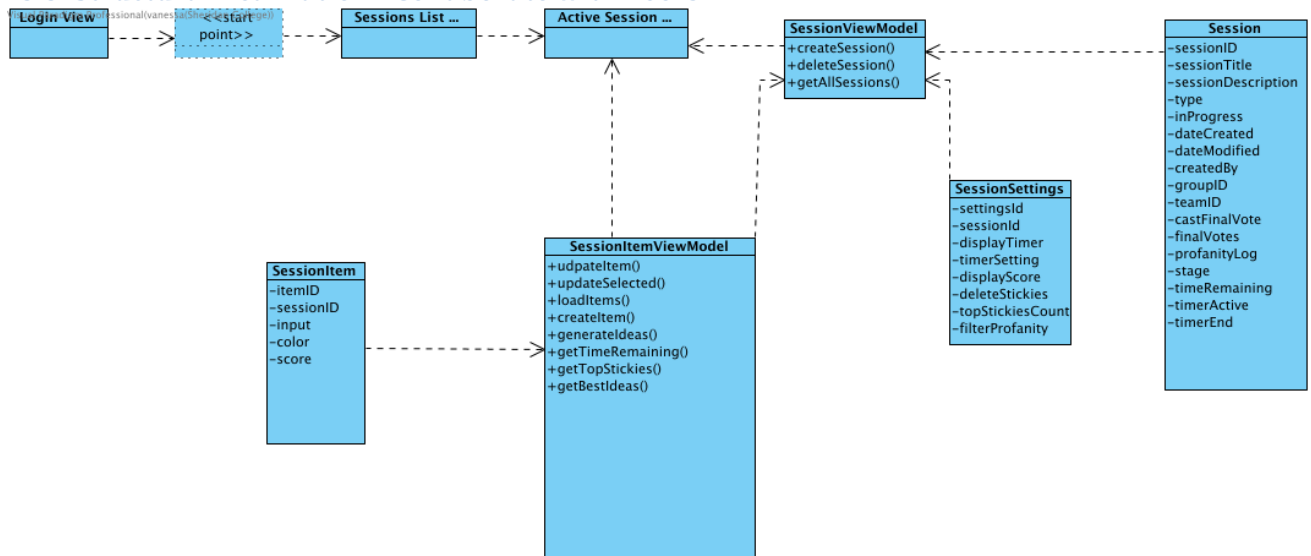


4.5.2. Collaboration Overview - Collaborate with Peers

Visual Paradigm Professional(vanessa(Sheridan College))



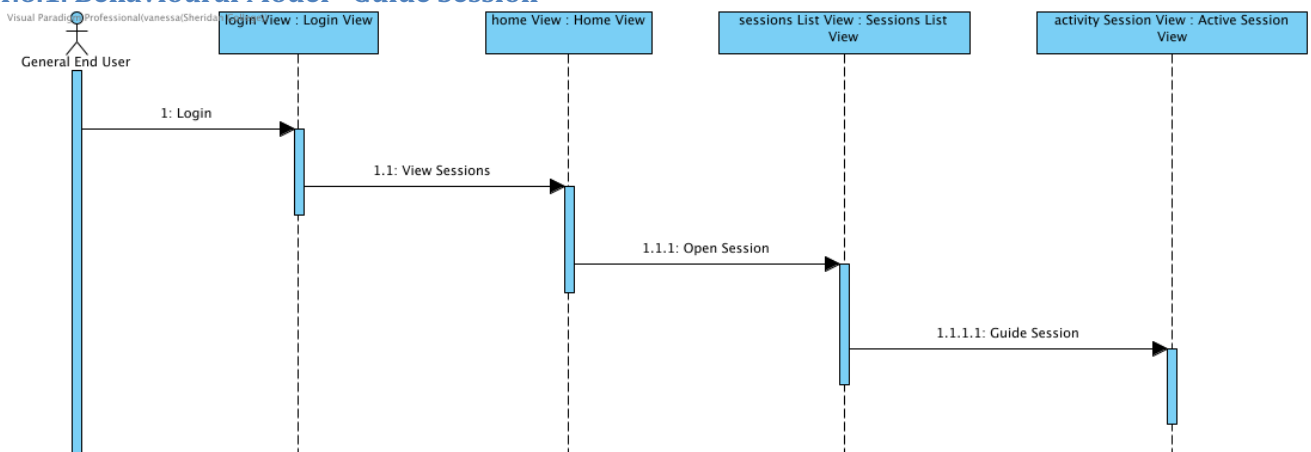
4.5.3. Structural Realization - Collaborate with Peers



4.6. Guide Session

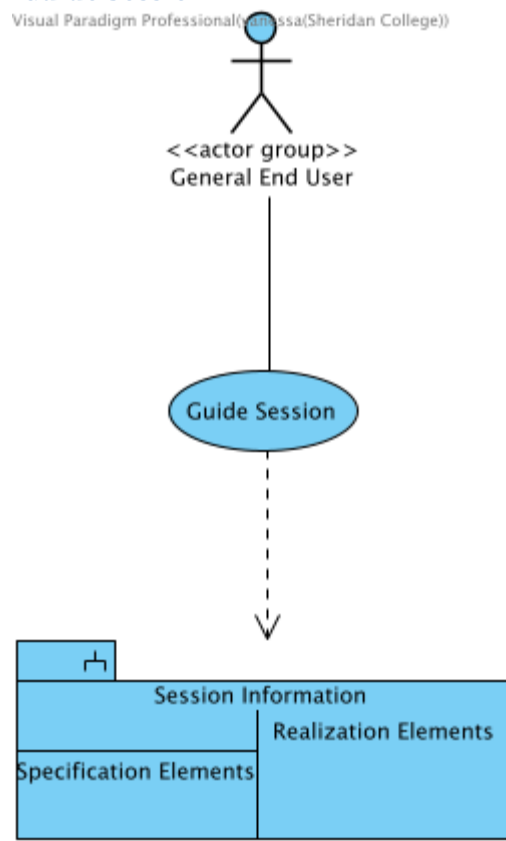
A tutorial will be set-up for the different session types to walk the user through the intended steps of the session

4.6.1. Behavioural Model - Guide Session



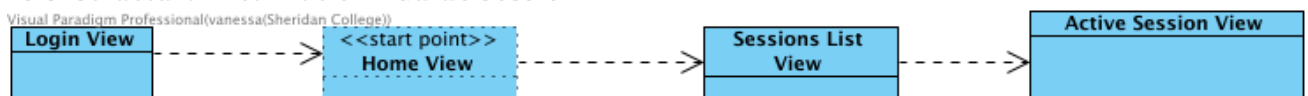
4.6.2. Collaboration Overview - Guide Session

Visual Paradigm Professional(vanessa(Sheridan College))



4.6.3. Structural Realization - Guide Session

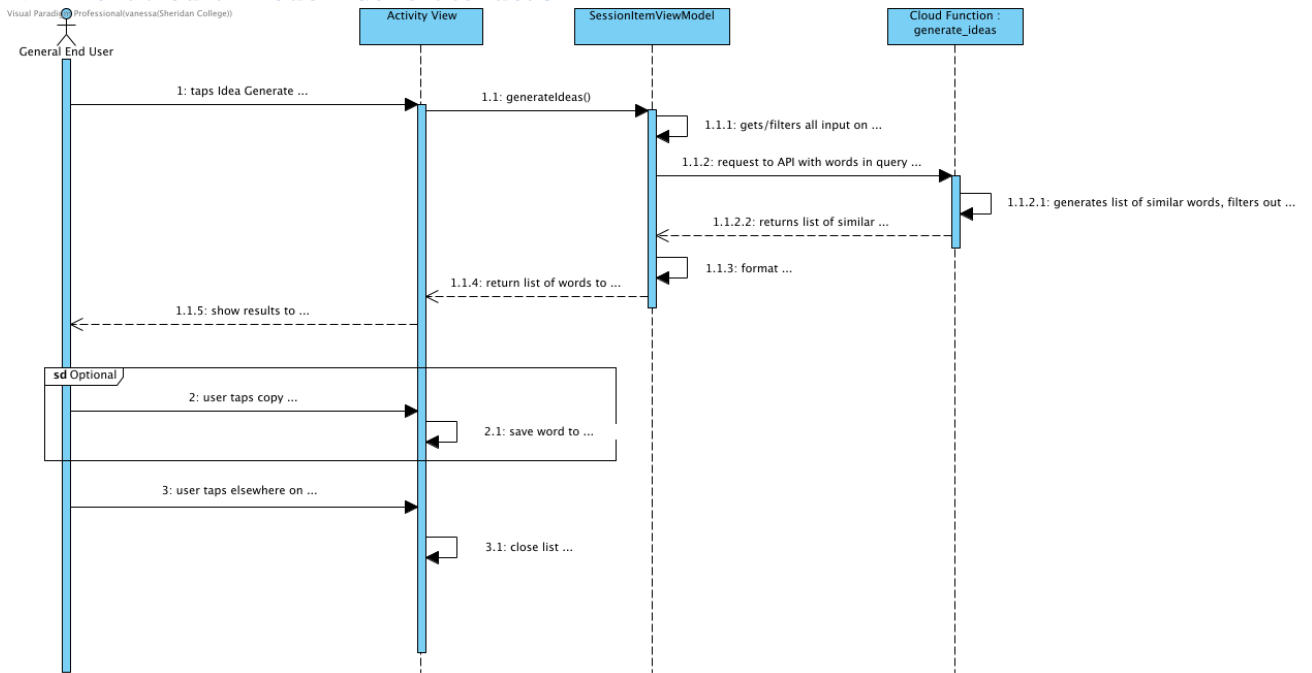
Visual Paradigm Professional(vanessa(Sheridan College))



4.7. Generate Ideas

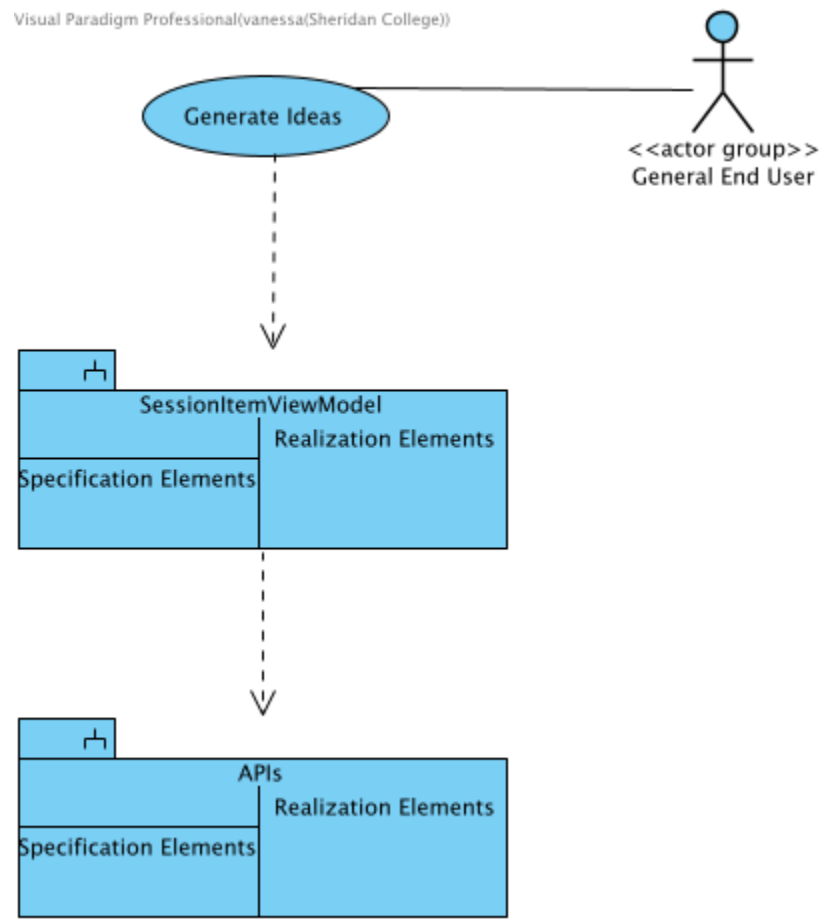
Users will be able to press a button within an activity that suggests a related word to them based on the words currently used in their session. This is intended to spark new thoughts.

4.7.1. Behavioural Model - Generate Ideas

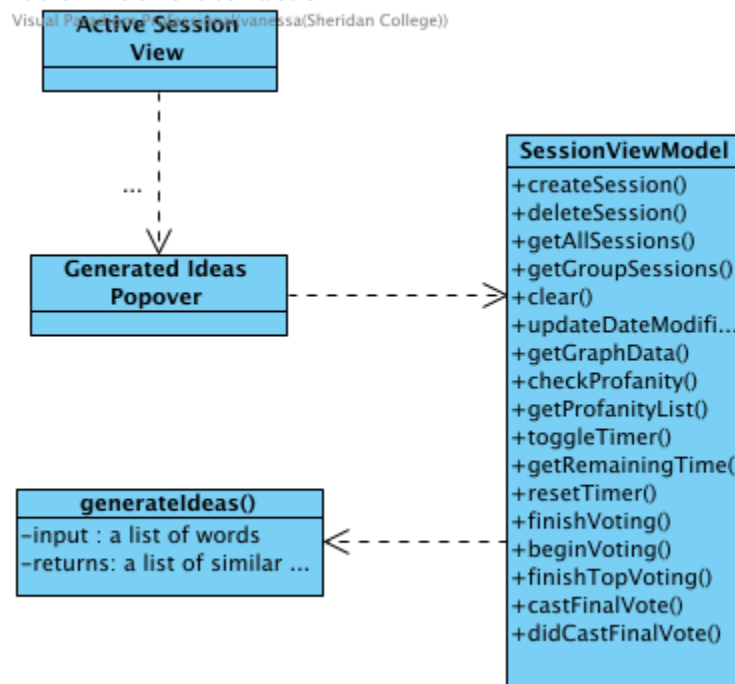


This model describes the behaviour of the Generate Ideas use case. This use case involves a user looking pressing a button that takes all user input on a workspace and generates/returns a list of new words related to the inputs.

4.7.2. Collaboration Overview - Generate Ideas



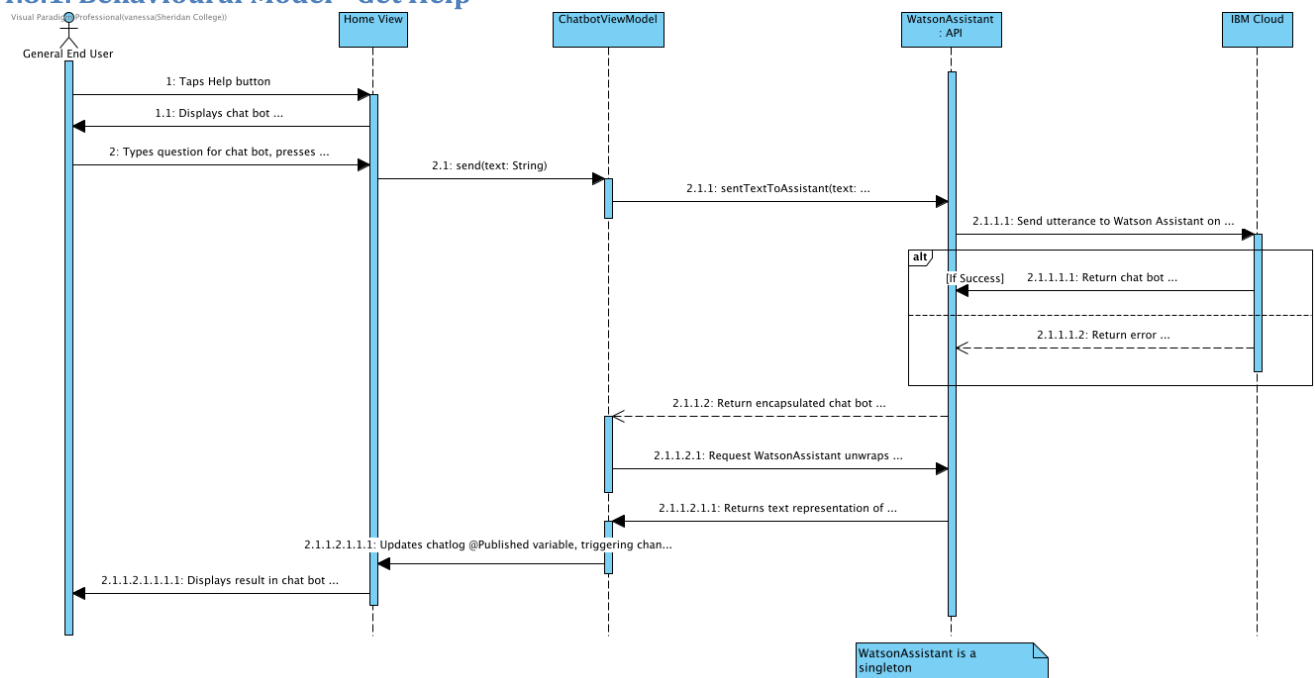
4.7.3. Structural Realization - Generate Ideas



4.8. Get Help

Users will be able to brainstorm with the help of an AI when using the app alone, as an optional feature that can be enabled in the session settings.

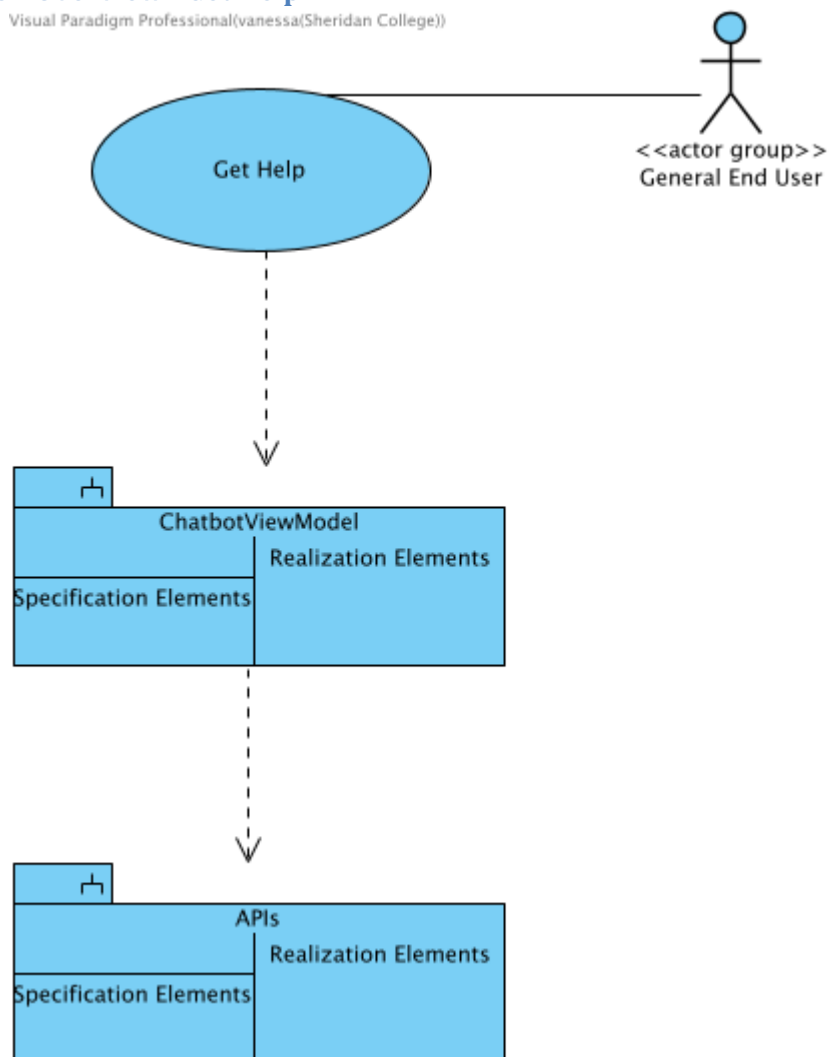
4.8.1. Behavioural Model - Get Help



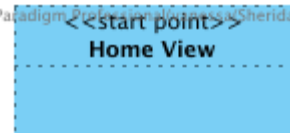
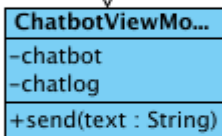
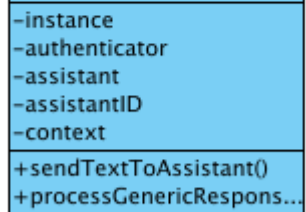
This model describes the behaviour of the Provide Help use case. This use case involves a user looking to get descriptive assistance via a chatbot using the application.

4.8.2. Collaboration Overview - Get Help

Visual Paradigm Professional(vanessa(Sheridan College))



4.8.3. Structural Realization - Get Help

**Help Button****WatsonAssistant**

From here, both operations within this API class are used to receive a response from the IBM Cloud Assistant