System and Software Architecture Description (SSAD)

Go Grrrls App

Team #03

Sl. No.	*Team Member*	*Roles*
1	**Anurag Tiwari**	**Prototyper/Implementer/Software Architect**
2	**Aravind Bhimarasetty**	**Operational Concept Engineer/ Implementer**
3	**David Tasky (Off-campus)**	**IIV & V/ Quality Focal Point**
4	**Jeevan Byappa Reddy**	**Life Cycle Planner/ Implementer**
5	**Pooja Doizode Raghavendra Rao**	**Project Manager/ UML Modeler**
6	**Sudhishna Sendhivelan**	**Requirements Engineer/ Implementer**
7	**Zhao Shi**	**Feasibility Analyst/ Builder**

12/05/2016

Version History

Date	Author	Version	Changes made	Rationale
10/11/2016	PD	1.0	• Added introduction and system analysis	• Initial draft for FC-package
10/17/2016	PD	1.1	• Modifications in use case diagram and artifact diagram	• Final FC- package
11/26/2016	PD	1.2	• Prototype Finalized – SSAD document update	• Updated the SSAD document in accordance with the finalized version of the prototype made for the app, after receiving client feedback and fixing all known bugs
12/05/2016	PD	2.0	• Added component, class and sequence diagrams, architecture styles and design patterns used in the project is described	• As Built Package

Table of Contents

Table of Tables

Table of Figures

1. Introduction

1.1 Purpose of the SSAD

The purpose of the SSAD is to demonstrate the details about the system software, architecture and hardware parts that will be used in the project. This report presents the key properties of the system by analyzing the system context diagram and showing the use cases.

1.2 Status of the SSAD

This is the version of SSAD has changes made based on the feedback given in FC review. It introduces the system context and use case diagrams for Go Grrrls project.

2. System Analysis

2.1 System Analysis Overview

The primary purpose of Go Grrrls app is to help teenage girls from the age of 10-14 years in the transition period from middle school to high school. Students enrolled in the Go Grrrls program learn about several areas of personal development. In order to help students, retain information and continue to engage with the Go Grrrls material, an iOS application will be developed and deployed in the app store. The app is formatted into chapters which allow the user to learn by reading and watching videos about common issues they face at their age. At the end of each chapter, there is an assessment quiz that the user can take which allows them to gage how well they understood the material.

2.1.1 System Context

Figure 1: System Context Diagram

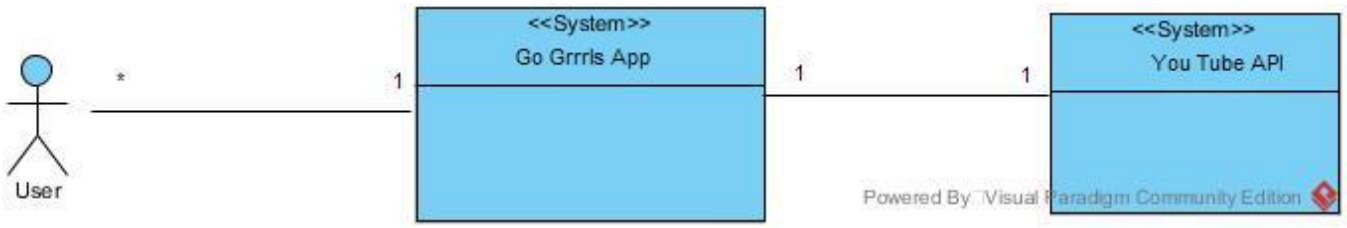


Table 1: Actors Summary

Actor	Description	Responsibilities
User	User of the system, here teenage girls from the ages of 10-14	We have only one kind of user in the system who is a teenage girl, therefore every use case is part of her responsibilities.

2.1.2 Artifacts & Information

Figure 2: Artifacts and Information Diagram

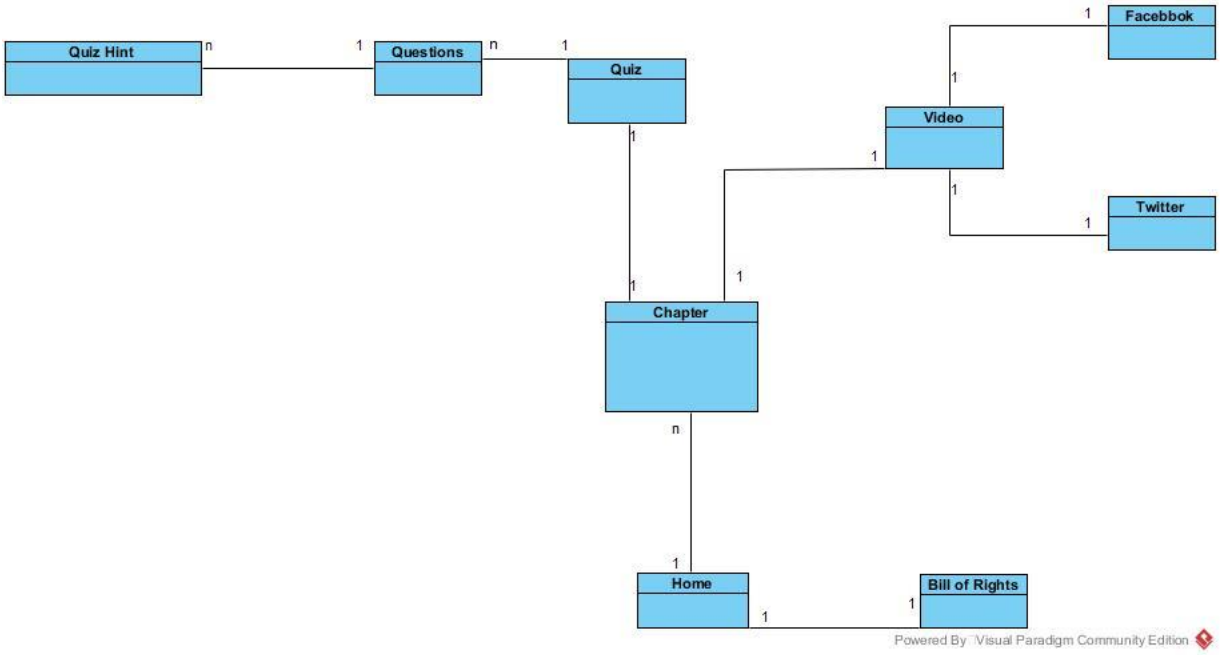
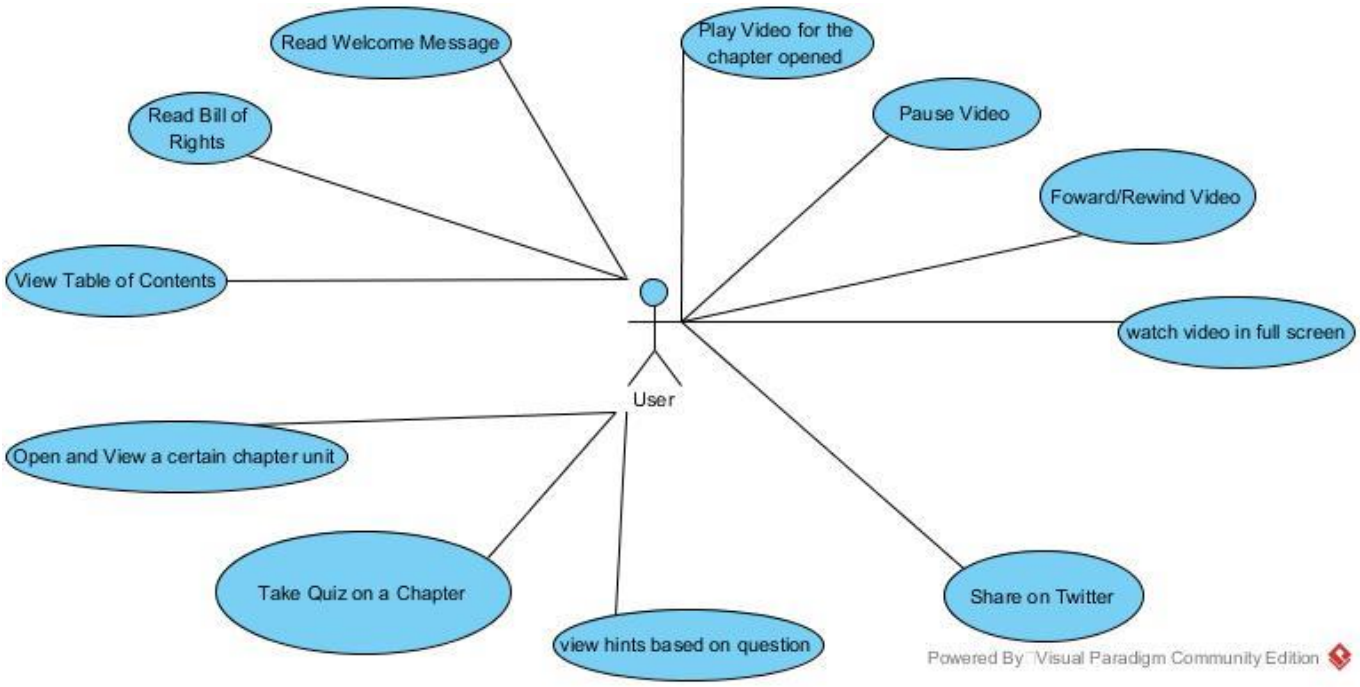


Table 2: Artifacts and Information Summary

Artifact	Purpose
Bill of Rights	Girl bill of rights shown to the user
Chapter	To display all the chapters to the user on the home of the app
Video	A video related to chapter for learning, user can pause, rewind/forward, stop and make a video full screen
Facebook	To share video on Facebook
Twitter	To post video on twitter
User	A teenage girl with Go Grrrls iOS app
Quiz	Assessment for the user at the end of each chapter
Quiz hint	Helpful tips provided to the user for wrong answers submitted

2.1.3 Behavior

Figure 3: Process Diagram



a. Process Welcome

Table 3: Process Description-Welcome

Identifier	UC-1: Welcome
Purpose	Welcome page for the users
Requirements	**WC_4290**: As a user, I should be able to view the welcome page of the Go Grrrls app upon clicking on the app icon.
Development Risks	None
Pre-conditions	None
Post-conditions	User is able to see the welcome page of Go Grrrls App

Table 4: Typical Course of Action-Welcome

Seq#	Actor's Action	System's Response
1	Open Go Grrrls App	
2		Show the welcome page on the app

b. Process Bill of Rights

Table 5: Process Description-Bill of Rights

Identifier	UC-2: Girl bill of rights
Purpose	Read bill of rights
Requirements	**WC_3963:**As a user, I can click and view the Girl Bill of Rights
Development Risks	None
Pre-conditions	None
Post-conditions	User is aware of girl bill of rights

Table 6 Typical Course of Action-Bill of Rights

Seq#	Actor's Action	System's Response
1	User opens Go Grrrls App	
2		Show the home page of the app
3	Click on view girl bill of rights	
4		Display the bill of rights

c. Process Table of Contents

Table 7 Process Description-Table of Contents

Identifier	UC-3: Table of Contents
Purpose	View all the chapters listed
Requirements	**WC_3928**: As a user, I should be able to see the list of all Chapters available on the Go Grrrls App.
Development Risks	None
Pre-conditions	None
Post-conditions	User can see the chapters listed for Go Grrrls App

Table 8 Typical Course of Action-Table of Contents

Seq#	Actor's Action	System's Response
1	User opens Go Grrrls App	
2		Show the home page of the app
3	Click on view content	
4		Display all the chapters listed

d. Process Chapter Open & View

Table 9 Process Description- Open and view a chapter

Identifier	UC-4: open and view a chapter
Purpose	View the chapter content
Requirements	**WC_4067**: As a user, I should be able to access the text contents of each chapter
Development Risks	None
Pre-conditions	None
Post-conditions	User is able to see the text content on that chapter

Table 10 Typical Course of Action- Open and view chapter

Seq#	Actor's Action	System's Response
1	User opens Go Grrrls App	
2		Show the home page of the app
3	Click on the chapter to view	
4		Display all the chapter text and other links available to go to from that chapter

e. Process Quiz System

Table 11 Process Description-Quiz System

Identifier	UC-5: Quiz system
Purpose	Assessment for the user at the end of each chapter
Requirements	**WC_4220:** As a user I should be able to assess myself by taking the tests in the App
Development Risks	None
Pre-conditions	None
Post-conditions	User can evaluate herself about the knowledge she gained from that chapter

Table 12 Typical Course of Action-Quiz System

Seq#	Actor's Action	System's Response
1	User opens Go Grrrls App	
2		Show the home page of the app
3	Click on view content	
4		Display all the chapters listed

5	User clicks on the chapter she would like to read	
6		Display the chapter content along with take quiz button at the bottom
7	User clicks on the quiz button	
8		Questions are given to the user
9	User submits the quiz	
10		Answers are recorded and gives hints to the users for the wrong answers

f. Process Quiz Hints

Table 13 Process Description-Quiz hints

Identifier	UC-6: Quiz hints
Purpose	To provide hints to the user in quiz in order to make them choose the right answer
Requirements	**WC_4088**: As a user, I should be able to see hints for the correct answers/see the correct answer, if I pick the wrong answer to a quiz question in the app.
Development Risks	None
Pre-conditions	The user choses wrong answer for a quiz question
Post-conditions	User is lead to choose the correct answer for the question

Table 14 Typical Course of Action-Quiz hints

Seq#	Actor's Action	System's Response
1	User opens Go Grrrls App	
2		Show the home page of the app
3	Click on view content	
4		Display all the chapters listed
5	User clicks on the chapter she would like to read	
6		Display the chapter content along with take quiz button at the bottom
7	User clicks on the quiz button	
8		Questions are given to the user
9	User answers the quiz question wrong	
10		Answers are recorded and gives hints to the users for the wrong answers

g. Process Play Video

Table 15 Process Description- Play Video

Identifier	UC-7: Play video from within the app
Purpose	Video embedded in line with text content for better understanding
Requirements	**WC_4221**: As a user I should be able to play YouTube video and audio from within the app
Development Risks	None
Pre-conditions	None
Post-conditions	User get more information about the chapter by watching the video

Table 16 Typical Course of Action- Play Video

Seq#	Actor's Action	System's Response
1	User opens Go Grrrls App	
2		Show the home page of the app
3	Click on view content	
4		Display all the chapters listed
5	User clicks on specific chapter	
6		Chapter content is loaded and with watch more link listed
7	User clicks on Watch more link	
8		The video is loaded from the YouTube API and is available to watch along with the text in-line
9	The user clicks on the play button	
10		The video starts playing

h. Process Pause Video

Table 17 Process Description- Pause Video

Identifier	UC-8: Pause the video
Purpose	Video to be paused if the user wants to read the text or wants to pause watching the video
Requirements	**WC_4087**: As a user, I should be able to play, pause and fast-forward a video or audio in the app.
Development Risks	None

Pre-conditions	The video embedded in the chapter is being played
Post-conditions	User gets to pause the video if she wishes to do so.

Table 18 Typical Course of Action- Pause Video

Seq#	Actor's Action	System's Response
1	User opens Go Grrrls App	
2		Show the home page of the app
3	Click on view content	
4		Display all the chapters listed
5	User clicks on specific chapter	
6		Chapter content is loaded and with watch more link listed
7	User clicks on Watch more link	
8		The video is loaded from the YouTube API and is available to watch along with the text in-line
9	The user clicks on the play button	
10		The video starts playing
11	The user clicks on the Pause button listed in the video window	
12		The video is paused

i. Process Forward/Rewind Video

Table 19 Process Description- Forward/Rewind Video

Identifier	UC-9: Forward/Rewind the video
Purpose	User may forward or rewind the video
Requirements	**WC_4087**: As a user, I should be able to play, pause and fast-forward a video or audio in the app.
Development Risks	None
Pre-conditions	The video embedded in the chapter is being played
Post-conditions	User gets to forward/rewind the video if she wishes fasten or slow down the speed of watching the video

Table 20 Typical Course of Action- Forward/Rewind Video

Seq#	Actor's Action	System's Response
1	User opens Go Grrrls App	
2		Show the home page of the app
3	Click on view content	
4		Display all the chapters listed
5	User clicks on specific chapter	
6		Chapter content is loaded and with watch more link listed
7	User clicks on Watch more link	
8		The video is loaded from the YouTube API and is available to watch along with the text in-line
9	The user clicks on the play button	
10		The video starts playing
11	The user clicks on the Forward/Rewind button listed in the video window	
12		The video is played faster or slowed down when forward and rewind button is clicked upon respectively

j. Process Share on Twitter

Table 21 Process Description- Share video on Twitter

Identifier	UC-10: Share video on Twitter
Purpose	User can share the video on social media
Requirements	**WC_4087**: As a user, I should be able to play, pause and fast-forward a video or audio in the app.
Development Risks	None
Pre-conditions	The video embedded in the chapter is being played
Post-conditions	User shares the video on her Twitter account

Table 22 Typical Course of Action- Share video on Twitter

Seq#	Actor's Action	System's Response
1	User opens Go Grrrls App	
2		Show the home page of the app
3	Click on view content	
4		Display all the chapters listed
5	User clicks on specific chapter	
6		Chapter content is loaded and with watch more link listed
7	User clicks on Watch more link	
8		The video is loaded from the YouTube API and is available to watch along with the text in-line. The layout displays the share option.
9	The user clicks on the share option	
10		The list of social media frameworks available is displayed
11	The user clicks on the Twitter option among the listed one	
12		User is taken to login page of Twitter
13	User enters her Twitter account credentials	
14		The video is posted on user's Twitter account

k. Process Video Full screen

Table 23 Process Description- Play video in full-screen

Identifier	UC- 11: Play video in full-screen
Purpose	Video made full-screen
Requirements	**WC_4086**: As a user, I should be able to consume in-app videos in-line with text and also on full-screen.
Development Risks	None
Pre-conditions	The video is loaded
Post-conditions	User can watch the video in full-screen mode

Table 24 Typical Course of Action- Play video in full-screen

Seq#	Actor's Action	System's Response
1	User opens Go Grrrls App	

2		Show the home page of the app
3	Click on view content	
4		Display all the chapters listed
5	User clicks on specific chapter	
6		Chapter content is loaded and with watch more link listed
7	User clicks on Watch more link	
8		The video is loaded from the YouTube API and is available to watch along with the text in-line
9	The user clicks on the full-screen option listed on the video layout window	
10		The video starts playing in full-screen mode

2.1.4 Modes of Operation

The Go Grrls iOS app does not involve more than one mode of operation.

2.2 System Analysis Rationale

We have only one kind of operational stakeholder. They are the teenage girls from the ages of 10-14 years and having an iOS device. Users will have to download the Go Grrrls App available on App store. There is no registration or login required to use the application. The user needs to have a reliable internet connection on their iOS device.

The application allows the user to view Girl Bill of Rights. The Go Grrrls application provides the users a list of chapters to read from, also for each chapter it shows the video related to that chapter in line with the text content. The system allows the user to watch video in full screen, play, pause, resume or stop it whenever she would like to. In addition to this at the end of each chapter user can take up an assessment test in the form of a quiz for which useful hints will be provided in case the user gets a question wrong.

In this way, the Go Grrrls application is designed to help the teenage girls to transition from middle school to high school.

3. Technology-Independent Model

This section has been intentionally omitted. Please refer to the technology-specific design is documented in the next section. The technology-independent model would be redundant for our system as the client has specified the technology to be used. Furthermore, the technology-independent model can easily be derived from the technology-dependent model.

4. Technology-Specific System Design

4.1 Design Overview

4.1.1 System Structure

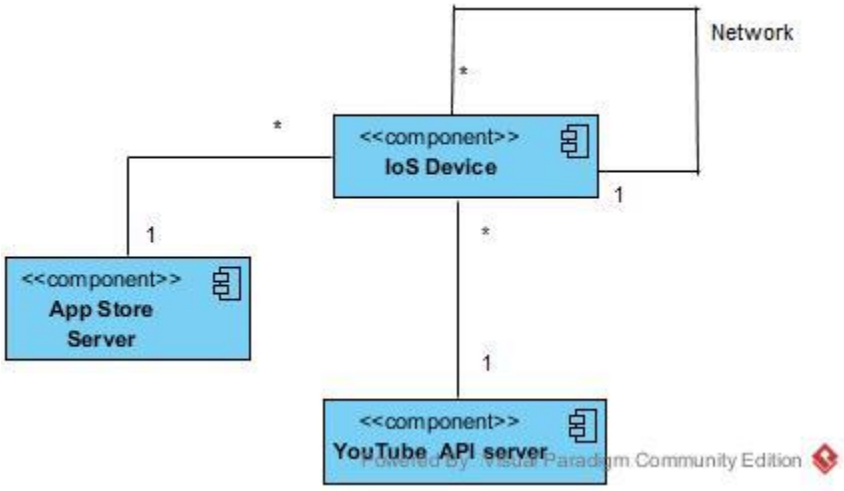


Figure 4: Hardware Component Class Diagram

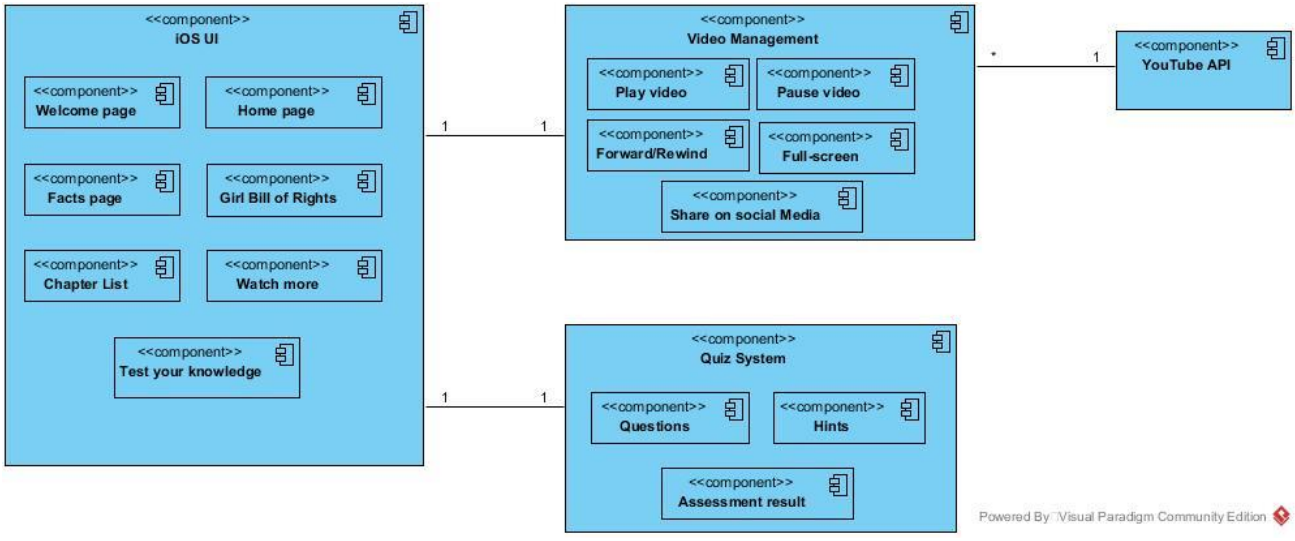


Figure 5: Software Component Class Diagram

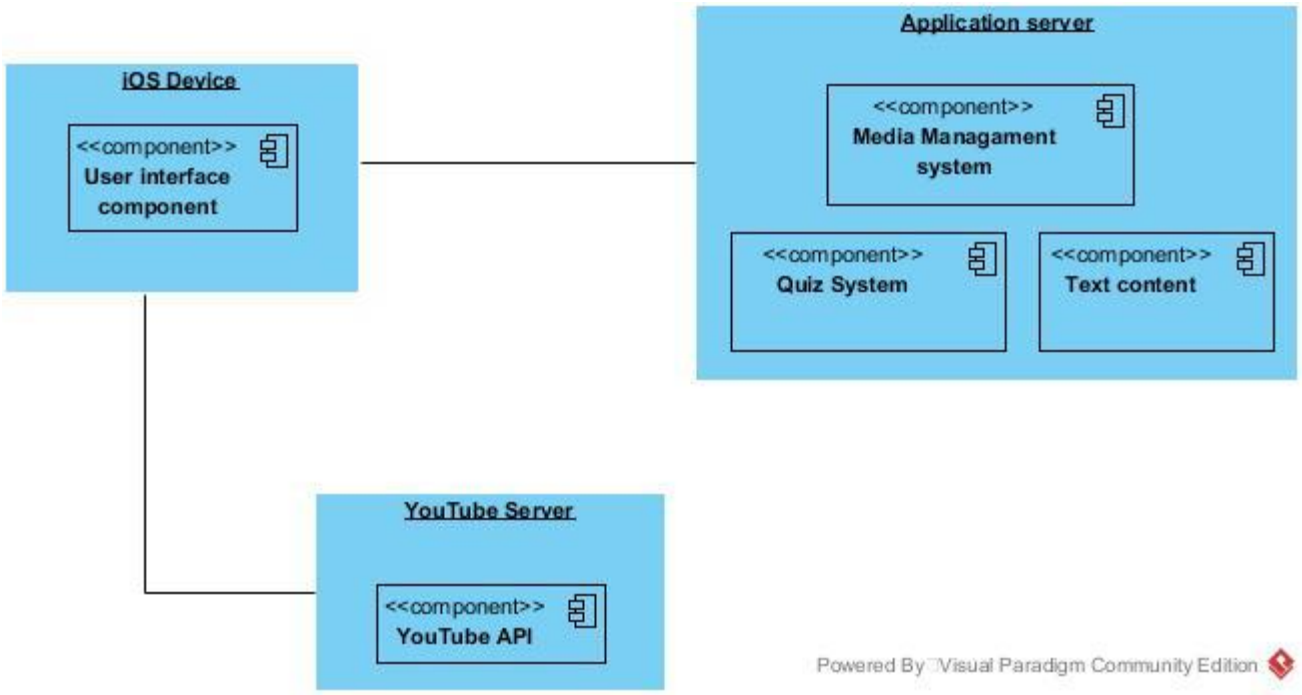


Figure 6: Deployment Diagram

Table 25 Hardware Component Description

Hardware Component	Description
iOS Device	Device used by the users to install and run app
YouTube API server	The server on which the videos are stored
App-Store server	The web server for App-Store on which the app deployed

Table 26 Software Component Description

Software Component	Description
Home Page	The home-screen of the page providing quick links to menu page, Bill of rights and Facts page
Menu page	The page listing all the chapters on the Go Grrrls app
Welcome page	The introductory page for the Go Grrrls App

Table 27 Supporting Software Component Description

Support Software Component	Description
Girl Bill of Rights	The page displaying the rights that girls should know they have
Watch more	Watch more component is for the users to watch videos regarding the chapter the user is reading
Test your knowledge	This component for the users to assess themselves at the

	end of each chapter.
Video Management	This component is used for video/audio management which includes embed video inline with text, play, pause, resume/forward video and share it on social media.
Quiz Management	This component consists of all the quiz questions, answers and the hints to them. It provides the user an assessment at the end of every chapter to test her knowledge.

4.1.2 Design Classes

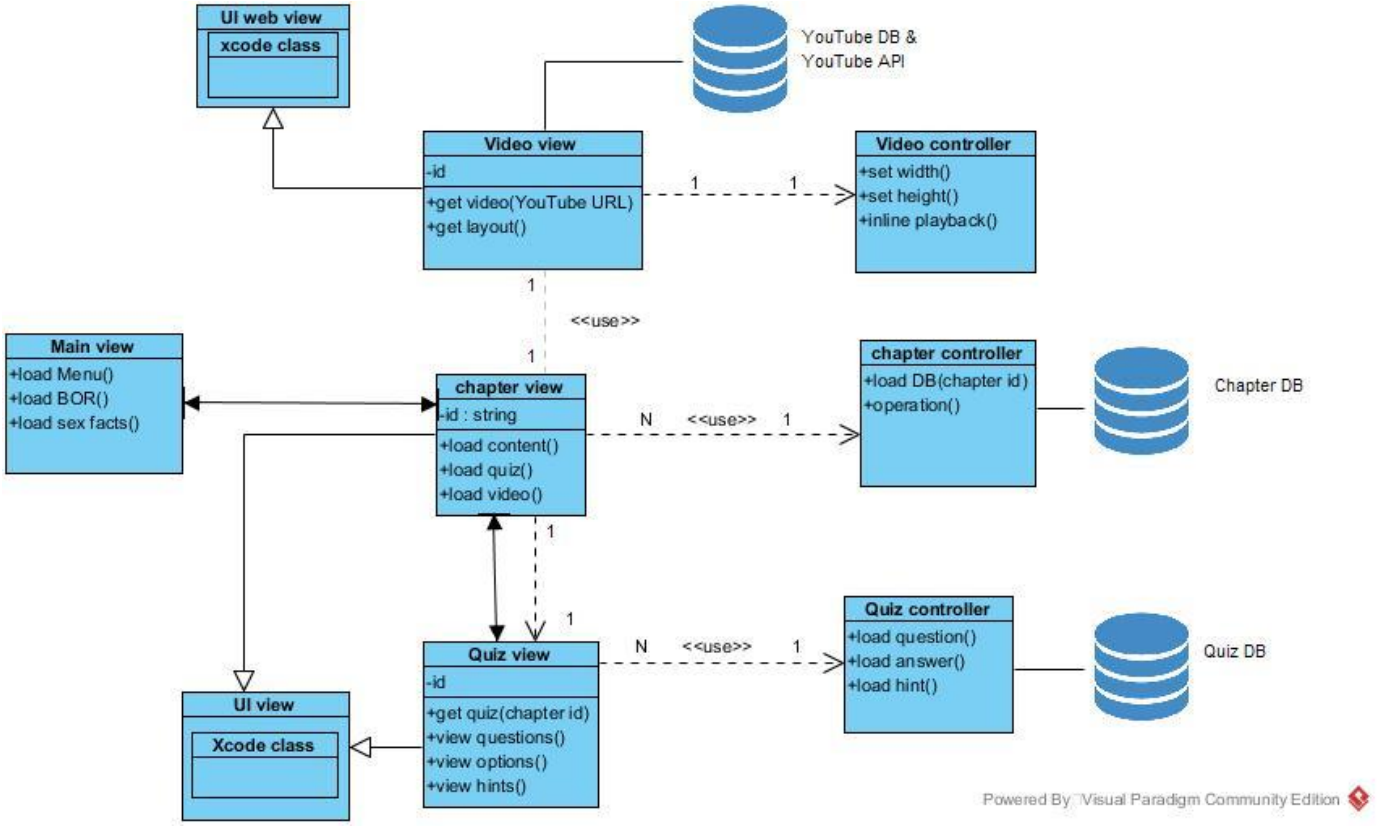


Figure 7: Design Class Diagram

Table 28: Design Class Description

Class	Type	Description
UIView	View	Main XCode view class that defines how views are loaded in the app(constraints,

		alignment, etc.)
UIWebView	View	Main XCode view class that defines how html web content is loaded in the app
MainView	View	View in the app that loads when the app starts. Contains links to chapter view and extra link views like bill of rights and facts pages
VideoView	View	View that uses webview structure to load YouTube videos in the app using embed html string.
VideoController	Controller	Controller that controls how Youtube videos are loaded into the VideoView – html link, embed constraints, boundary & padding, video controls, orientation override and inline playback constraint.
ChapterView	View	View that loads the scrollable and attributed text content and video in the app
ChapterController	Controller	Controller that controls how text content is loaded in the app and controls layout boundary constraints for loading the video view as part of the chapter view.
QuizView	View	View that loads question(labels), answers(buttons) and hints(textview) along with end of quiz message and results (textview)
QuizController	Controller	Controller that controls how the questions, answers and hints are loaded into the quiz view and when and according to which user input they should be loaded into the view.

4.1.3 Process Realization

In this section, we have presented the sequence diagrams for the core capabilities of the Go Grrrls App.

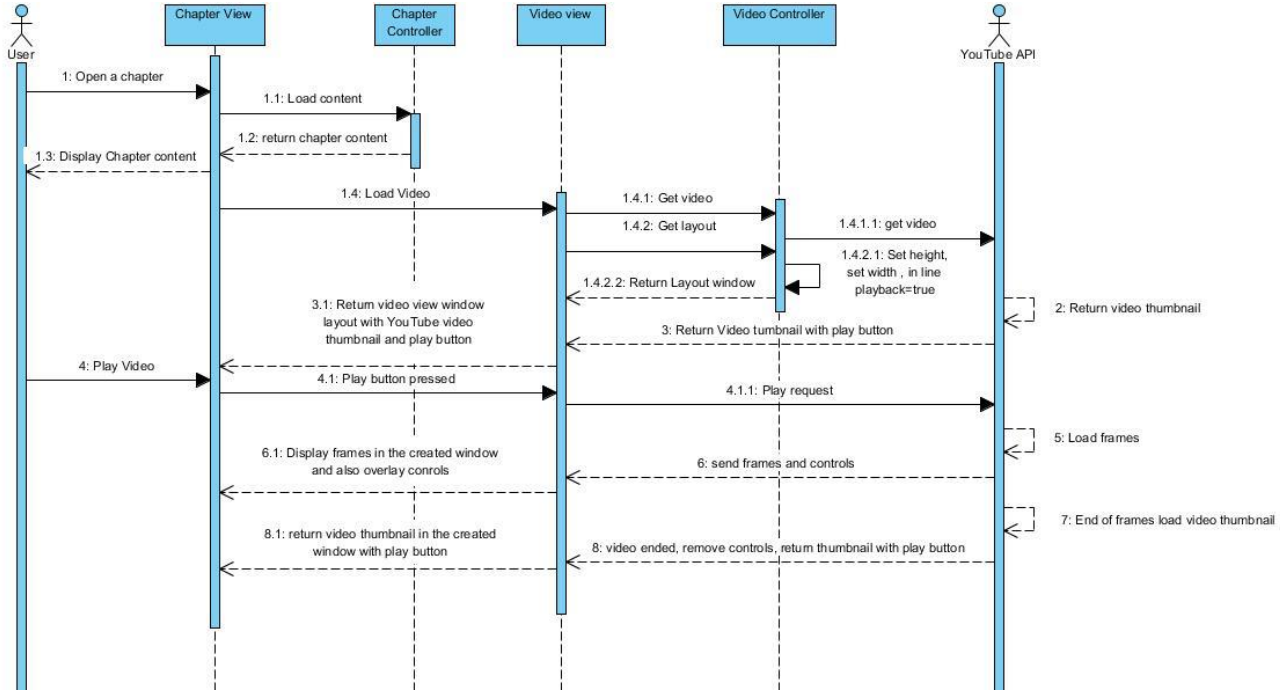


Figure 8: Video Management

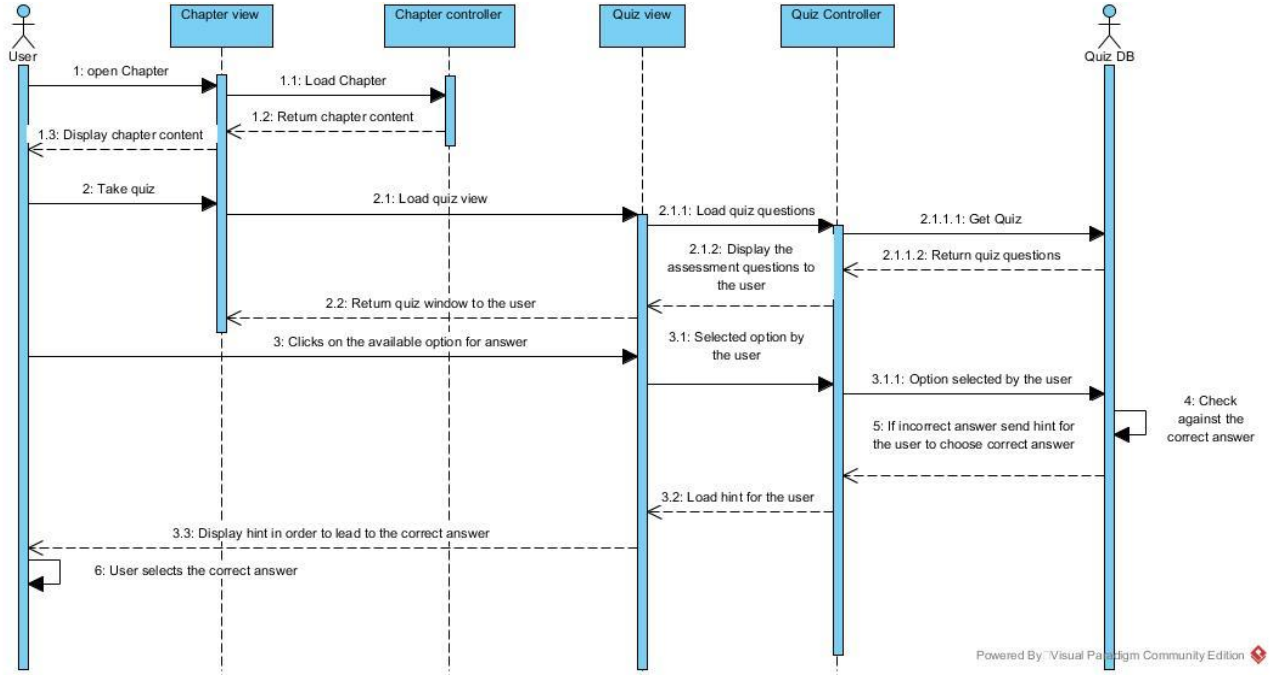


Figure 9 Quiz System

4.2 Design Rationale

Go Grrrls App will be used by teenage girls from the age of 10 - 14 who are enrolled in the Go Grrrls program, which is an after school program offered in some middle school/high schools in Arizona. The app will be an open iOS app on the Apple AppStore for everyone since the Go Grrrls Program is a non-profit program. Girls, both enrolled in the Go Grrrls program & outside the program will learn about several key areas of personal development by using this App.

- ❖ This iOS app follows MVC (Model View Controller) architecture (see section 5 – Architectural Styles, Patterns and Frameworks). This is done to help development using the Cocoa application package for iOS, which itself follows the MVC structure. This essentially splits up the system in 3 parts
  - The view (App User Interface)
    - iOS device screens
  - The Controller (functional logic controlling the views)
    - AV loading
    - Quiz System
    - User Input response controller
  - The Model (raw data)
    - Content of the app (text and video links)
      - Chapter Text
      - Quiz questions, answers and hints
      - YouTube embed links

- Audio/Visual Content Hosting on YouTube – The app contains chapters which in turn contains videos that provide a rich content consumption experience for the user. These videos are large due to the high resolutions and frame rate (1080p30fps). Thus, the decision was taken to use YouTube, a video hosting and consumption platform from Google, instead of hard-embedding the videos into the app which would result in slow response and loading time for the app.

5. Architectural Styles, Patterns and Frameworks

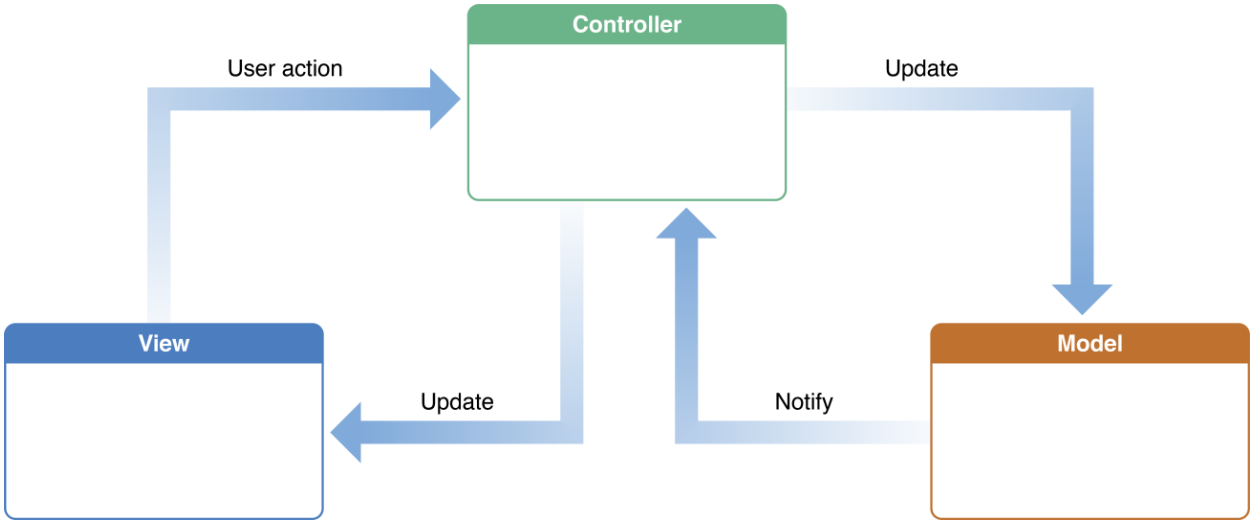


Figure 10 MVC Structure

Image Source: https://developer.apple.com/library/content/documentation/General/Conceptual/DevPedia-CocoaCore/MVC.html

Table 29: Architectural Styles, Patterns, and Frameworks

Name	Description	Benefits, Costs, and Limitations
MVC (Model View Controller) architectural style for designing the app.	It is a popular (if not the only) design pattern for development of iOS applications. It is made up of three components namely, model- lowest level responsible for maintaining data, view- responsible for displaying data to the user and getting user input, controller- functional logic which controls interaction between model and view. The Go Grrrls iOS app is implemented using Cocoa application package on XCode which follows MVC architecture.	Benefits • The UI is better defined and more specific to device • Allows for dynamic changes in the app making it easily adaptable to changing environments (e.g. iOS app can easily be made into desktop MacOsX app) • Objects, in code (models and controllers), are reusable (across multiple views) Costs • Free Limitations • Increased complexity • Closely coupled view and controller makes modifications to one affect the other • An active model can cause excessive updates of the corresponding view, which can result in a laggy and less responsive final product. **The model (data-text and video links) for our app is static.*
YouTube Iframe API for hosting video content in the app.	This Iframe JavaScript structure allows for HTML NS-String based callbacks to embed Videos in iOS app developed on SWIFT or Objective-C programming language. This structure allows full control of the videos (borders/padding, frame cache and dpi controls) through simple HTML callbacks to the YouTube server.	Benefits • Easily bridges the gap between application SWIFT code and YouTube player's JavaScript code. • Allows the app to be lightweight since the videos are not hard-embedded into the app. Costs • Free Limitation • Frequent updates to YT's API can cause issues in callbacks since it's an open source project • Device would require a data package connection(internet) to load and watch the video.