

Assignment 1: Mobile Application Design Specification

Game Trooper

Ray Krishardi Layadi - 26445549

Application Concept

Introduction

Game Trooper is an interactive mobile application that delivers news about games from variety of game news websites. This application plans to achieve three main objectives which revolve around user experience and entertainment:

The first main objective of this application is to allow users to stay up-to-date with the current popular new games as well as the new upcoming games by providing several articles and news from various gamers' trusted websites. Top charts of these current popular and upcoming games would also be provided by this application which would ease users in finding trending games for their beloved gaming platforms.

The second main objective of this application is to assist users in finding local game stores in Melbourne to help them buy their anticipated games in a reliable and efficient manner.

The third main objective of this application is to provide entertainment to users by allowing them to play a fun True or False quiz game. Questions are prepared based on popular game titles from various game genres.

In addition to the three main objectives of this application, there are additional features that enhance the user experience when using this application. These include speech recognition feature when searching for a game store and support for multiple gestures.

Target Audience

The target audience for this application include children, adolescence, young adult, and middle aged who are passionate about gaming. However, the primary target audience for this application is adolescence gamers. Therefore, several factors should be considered:

1. Pricing – The application is free as paid applications are not appealing to adolescence. This is mainly because there are several free alternative applications that also deliver the same main functionalities.
2. User Interface – As the primary target audience is adolescence gamers, the user interface would use consistent coloring instead of using multiple coloring (i.e. not mixing too much color in a screen). In addition, the design of the user interface would conform to the Android and iOS design guidelines. It would provide accessibility for different type of users while maintaining simplicity. As a result, the application would cater for users with different ages and technical expertise.
3. Mobile Operating System – The selection of mobile operating system would affect how well the application would be downloaded by various users. Therefore, Android and iOS are selected as the targeted mobile operating system.

Application Functionality

- The ability to display game news from variety of game news websites
 - The ability to display more detailed information about the selected game news
- The ability to display the top 10 current and upcoming games including all relevant information (e.g. available platform, review score, etc.)
 - The ability to display more detailed information about the selected game (e.g. game description, genre, developer, publisher, release date, etc.)
- The ability to pin local game stores in Melbourne on maps
 - The ability to track user's current location
- The ability to search for game stores
 - The ability to perform speech recognition and find the desired game stores
- The ability to play True or False quiz game
 - The ability to display the main game screen
 - The ability to play the game when the "Play" button is tapped
 - The ability to play the game
 - The ability to display the question
 - The ability to display the user response mechanism
 - Arrow that will move to either left or right when the user move/tilt their mobile device accordingly (able to detect motion of a device and execute the intended action)
 - Located between the green check mark and the red cross mark
 - Comprised of three stages
 - Stage 1 = 1 arrow
 - Stage 2 = 2 arrows
 - Stage 3 = 3 arrows
 - Green check mark and red cross mark that represent true or false
 - Green check mark represents true and located on the left side of the arrow
 - Red cross mark represents false and located on the right side of the arrow
 - Next button to move to the next question and end the game when all questions have been answered
 - The ability to display the game progress (include indicator whether the answered question is correct or incorrect (represented by the small green check mark and red cross mark))
 - The ability to display the correct answer and provide details about the answer for each question

- The ability to display the game over screen
 - The ability to replay the game when the “Replay” button is tapped
 - The ability to display all correct and incorrect answered questions (represented by the small green check mark and red cross mark)
- The ability to support multiple gestures (e.g. swipe, double tap for smart zoom, and pinch to zoom in and zoom out) when the user is reading a particular game news or game details

Idea Innovation

Despite the fact that there are several existing mobile applications on the market that also deliver news about games, most of them are mobile application developed by the corresponding game news websites (e.g. IGN, GameSpot, etc.). As a result, news about games are only taken from the corresponding primary website. Therefore, “Game Trooper” could be considered as innovative for these type of mobile applications as it provides users with the ability to select and read news from multiple game news websites. In addition, this application also enables users to find local game stores in Melbourne which provides convenience in buying anticipated games. Furthermore, this application also provides entertainment to users by allowing them to play a fun True or False quiz game.

“Game Trooper” also utilizes mobile technology features to boost user experience (i.e. gyroscope, accelerometer, gesture recognizers, etc.).

1. Gyroscope and accelerometer would be utilized to detect device movement in True or False quiz game
2. Gesture recognizers would be utilized to provide support for multiple gestures (e.g. swipe, double tap for smart zoom, and pinch to zoom in and zoom out)

Android and iOS Technology Considerations

Web Networking

- The ability to display game news from variety of game news websites
 - The ability to display more detailed information about the selected game news
- The ability to display the top 10 current and upcoming games including all relevant information (e.g. available platform, review score, etc.)
 - The ability to display more detailed information about the selected game (e.g. game description, genre, developer, publisher, release date, etc.)

	Android	iOS
Required technology	<ol style="list-style-type: none">1. API from game news websites including the API Key2. RSS Feeds from game news websites3. Could choose one of the following technology:<ol style="list-style-type: none">3.1. HTML and HTML Parser (java.io.BufferedReader, java.io.InputStream, java.io.InputStreamReader, java.io.StringWriter, java.net.HttpURLConnection, java.net.URL)3.2. JSON and JSON Parser (org.json.*)	<ol style="list-style-type: none">1. API from game news websites including the API Key2. RSS Feeds from game news websites3. Could choose one of the following technology:<ol style="list-style-type: none">3.1. HTML and HTML Parser (UIKit)3.2. JSON and JSON Parser (UIKit or UIKit + Alamofire)
Method of implementation	Process and parse the JSON object obtained by using the game news website API	Process and parse the JSON object obtained by using the game news website API.
Verdict	Both mobile platforms could be used to implement this feature. However, it appears that the iOS implementation would probably require less code and save more time (only require the default framework UIKit) as opposed to the Android implementation.	

Maps and Geolocation

- The ability to pin local game stores in Melbourne on maps
 - The ability to track user's current location

	Android	iOS
Required technology	<ol style="list-style-type: none">1. Google Maps API2. Google Maps Activity which include all the required frameworks/libraries	<ol style="list-style-type: none">1. Apple Maps2. MapKit Framework3. CoreLocation Framework
Method of implementation	<p>Pin local game stores in Melbourne on Google Maps:</p> <ol style="list-style-type: none">1. Create Google Maps Activity2. Open the URL link given in the “google_maps_api.xml” in your browser3. Login with Google Account4. Create Android API Key5. Copy the key and replace the “YOUR_KEY_HERE” literal with your key in the “google_maps_api.xml”6. Get the latitude and longitude of each game stores in Melbourne7. Edit the MapsActivity.java to pin each game stores in Melbourne on Google Maps using the corresponding latitude and longitude8. Maintain a list of game stores addresses that correlate with the corresponding latitude and longitude	<p>Pin local game stores in Melbourne on Apple Maps:</p> <ol style="list-style-type: none">1. Add “Map Kit View” to the main storyboard and get reference to it in the view controller2. Import MapKit in your view controller3. Create annotation with the appropriate title and coordinate for each game stores in Melbourne4. Add annotations to the Apple Maps for display

	Android	iOS
	<p>Track user's current location:</p> <ol style="list-style-type: none"> 1. Add the <code>uses-permission</code> element with the appropriate location permission to the <code>"AndroidManifest.xml"</code> so that we can ask the user for permission to access current location 2. Create the appropriate location manager and listener that logs the user's location 3. Request permission for user location 4. If permission is granted then get a timely update on user's location 5. Keep track of the user's current location and display it on the Google Maps 	<p>Track user's current location:</p> <ol style="list-style-type: none"> 1. Add CoreLocation framework 2. Create and setup the location manager properly 3. Request permission for user location 4. If permission is granted then get a timely update on user's location 5. Keep track of the user's current location and display it on the Apple Maps
Verdict	Both mobile platforms could be used to implement this feature. However, it appears that the iOS implementation would probably require less code and save more time as opposed to the Android implementation. This is because you save time by not having to create and setup the Android API Key. In addition, it requires less code to achieve the same functionality.	

Web Networking and Audio Processing

- The ability to search for game stores
 - The ability to perform speech recognition and find the desired game stores

	Android	iOS
Required technology	<p>Search for game stores:</p> <ol style="list-style-type: none">1. Google Places API <p>Speech recognition:</p> <ol style="list-style-type: none">1. SpeechRecognizer library (<code>android.speech.SpeechRecognizer</code>)2. Require actual device	<p>Search for game stores:</p> <ol style="list-style-type: none">1. MapKit framework <p>Speech recognition:</p> <ol style="list-style-type: none">1. Speech framework2. Require actual device
Method of implementation	<p>Search for game stores:</p> <ol style="list-style-type: none">1. Setup Google Places API2. Implement the search game store function using the functionalities given by the Google Places API and display the result using the appropriate search view activity <p>Speech recognition:</p> <ol style="list-style-type: none">1. Request permission to access microphone and record audio2. Create a SpeechRecognizer3. Add the corresponding RecognitionListener to the SpeechRecognizer4. Invoke startListening method on SpeechRecognizer5. Get the string result of what being said6. Use the string result to search for the desired game store using Google Places API and display result using the appropriate search view activity	<p>Search for game stores:</p> <ol style="list-style-type: none">1. Implement the search game store function using <code>MKLocalSearchRequest</code> function from MapKit framework and display the result using <code>UISearchController</code> <p>Speech recognition:</p> <ol style="list-style-type: none">1. Request permission to access microphone and performing speech recognition2. Setup audio session to record audio3. Create a speech recognition request object on what being said on the microphone4. Check for audio input (recording) using the device's audio engine5. Add audio input (recording) to the speech recognition request6. Create a speech recognition task based on the speech recognition request

	Android	iOS
		<ol style="list-style-type: none">7. Get the string result of what being said8. Prepare and start the audio engine9. Use the string result to search for the desired game store using the MKLocalSearchRequest and display result using UISearchController
Verdict	Both mobile platforms could be used to implement this feature. However, further analysis needs to be conducted to determine the advantages and disadvantages of both mobile platforms.	

Mobile Sensors and Visual Processing

- The ability to play True or False quiz game

	Android	iOS
Required technology	<ol style="list-style-type: none">1. SQLite Database2. Sensor, Sensor Event, Sensor Manager, SensorEventListener, and TriggerEvent Listener3. Require actual device	<ol style="list-style-type: none">1. CoreData and Foundation framework2. CoreMotion3. Require actual device
Method of implementation	<p>Store questions and answers:</p> <ol style="list-style-type: none">1. SQLite Database will be used to store 15 quiz questions and answers <p>Play/Replay the game when the “Play”/”Replay” button is tapped:</p> <ol style="list-style-type: none">1. Create intent to pass data from one activity to another2. Inflate the appropriate layout when the “Play”/”Replay” button is tapped <p>Display questions, correct answer, and answer details:</p> <ol style="list-style-type: none">1. TextViews <p>Display next button:</p> <ol style="list-style-type: none">1. Button <p>Display game progress and all correct and incorrect answered questions:</p> <ol style="list-style-type: none">1. ImageView <p>User response mechanism:</p> <ol style="list-style-type: none">1. Add left and right arrow, green check mark, and red cross mark images	<p>Store questions and answers:</p> <ol style="list-style-type: none">1. CoreData will be used to store 15 quiz questions and answers <p>Play/Replay the game when the “Play”/”Replay” button is tapped:</p> <ol style="list-style-type: none">1. Create a segue between the “Play”/”Replay” button and the corresponding ViewController2. The segue will pass data from one view controller to another and display the appropriate view <p>Display questions, correct answer, and answer details:</p> <ol style="list-style-type: none">1. Labels <p>Display next button:</p> <ol style="list-style-type: none">1. Button <p>Display game progress and all correct and incorrect answered questions:</p> <ol style="list-style-type: none">1. ImageView <p>User response mechanism:</p> <ol style="list-style-type: none">1. Add left and right arrow, green check mark, and red cross mark images

	Android	iOS
	<p>with the appropriate positioning and in the appropriate layout</p> <ol style="list-style-type: none"> 2. Setup accelerometer and gyroscope sensors to monitor device's movement using the sensor framework 3. Move the arrow to the left or right accordingly based on the accelerometer and gyroscope sensors data 4. When the arrow has reached either the green check mark or red cross mark then determine whether the user answer is correct or incorrect 5. After answering 5 quiz questions, the game is over. When the game is over, user will be asked whether they want to replay the game or not. In addition, the screen will display how many questions user has answered correctly. 	<p>with the appropriate positioning and in the appropriate ViewController</p> <ol style="list-style-type: none"> 2. Setup accelerometer and gyroscope sensors to monitor device's movement using the CoreMotion framework 3. Move the arrow to the left or right accordingly based on the accelerometer and gyroscope sensors data 4. When the arrow has reached either the green check mark or red cross mark then determine whether the user answer is correct or incorrect 5. After answering 5 quiz questions, the game is over. When the game is over, user will be asked whether they want to replay the game or not. In addition, the screen will display how many questions user has answered correctly.
Verdict	Both mobile platforms could be used to implement this feature. However, further analysis needs to be conducted to determine the advantages and disadvantages of both mobile platforms.	

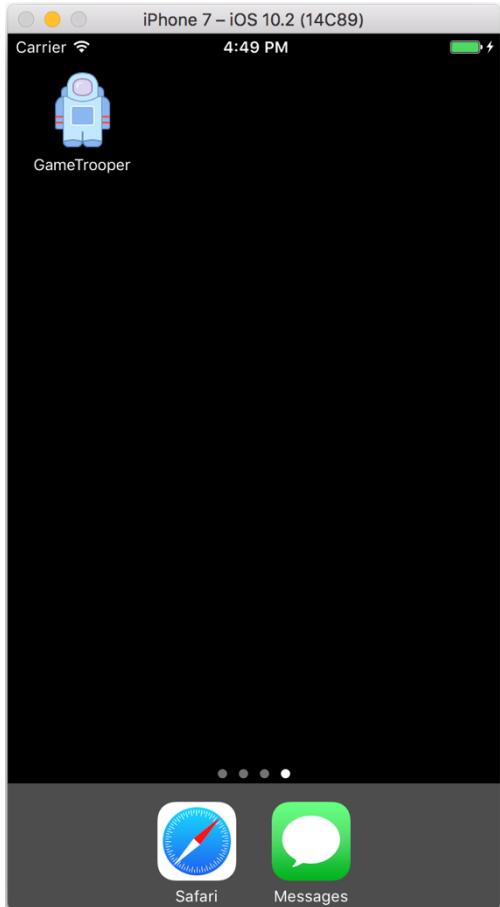
Gestures

- The ability to support multiple gestures (e.g. swipe, double tap for smart zoom, and pinch to zoom in and zoom out) when the user is reading a particular game news or game details

	Android	iOS
Required technology	1. GestureDetector, GestureDetectorCompat, and MotionEvent (<code>android.view.GestureDetector</code> , <code>android.support.v4.view.GestureDetectorCompat</code> , <code>android.view.MotionEvent</code>)	1. Gesture Recognizers (UIKit framework)
Method of implementation	1. Implement the <code>GestureDetector.OnGestureListener</code> and <code>GestureDetector.OnDoubleTapListener</code> in the appropriate activity class 2. Create <code>GestureDetectorCompat</code> object with the appropriate Context and <code>GestureDetector.OnGestureListener</code> 3. Set the <code>OnDoubleTapListener</code> to the <code>GestureDetectorCompat</code> object 4. Add the appropriate gesture functionalities to each overridden gesture method and make sure each of the gesture method return true	1. Create the appropriate gesture recognizer object which targets to a particular ViewController and specify the appropriate action/function to be executed when that gesture is performed 2. Add the gesture recognizer object to the view's gesture recognizer
Verdict	Both mobile platforms could be used to implement this feature. However, it appears that the iOS implementation would probably require less code and save more time as opposed to the Android implementation.	

Android and iOS Interface Design Storyboard Mockups

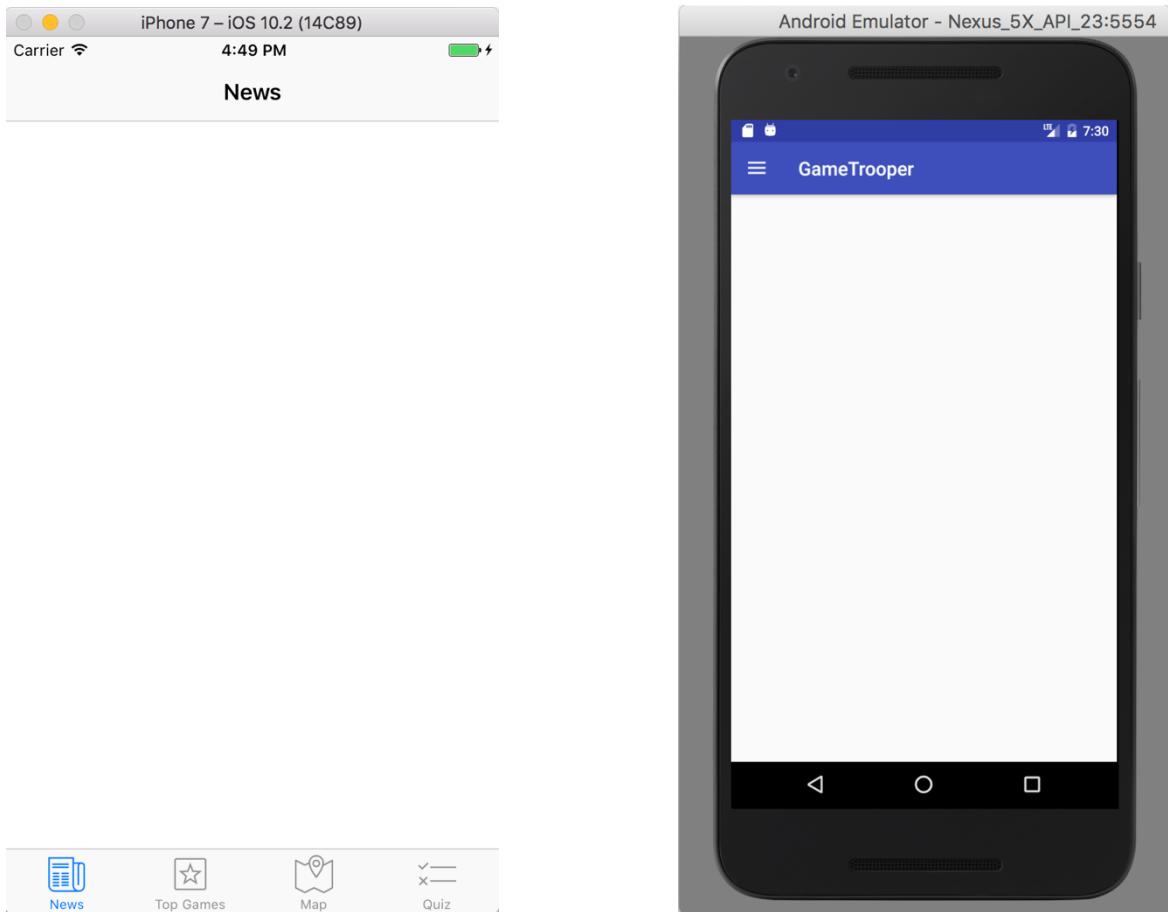
1. Application icon



Description

Both Android and iOS will have the same application icon (astronaut).

2. Launch/Splash screen

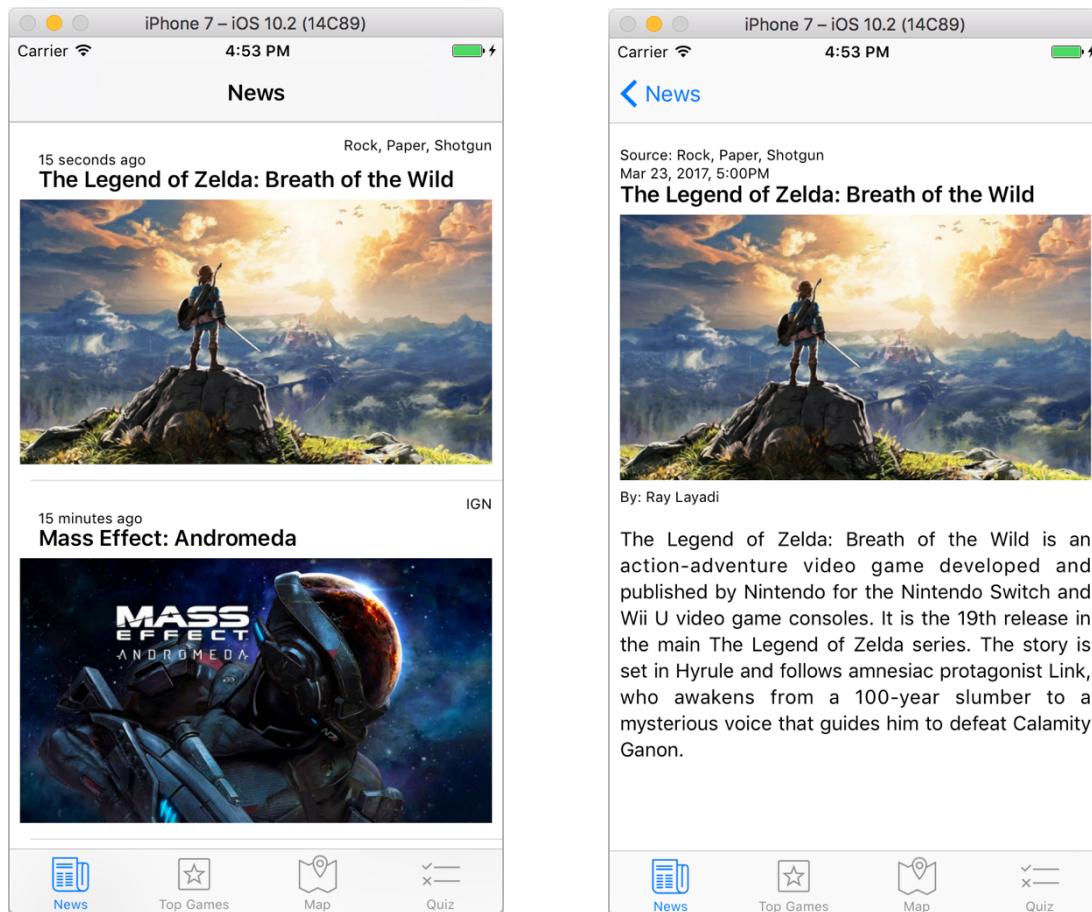


Description

As mentioned in the iOS Human Interface Guidelines and Android Material Design Guidelines, a launch screen is required to enhance the user's first impression of our application by providing the notion that the application is fast and responsive. In addition, it is mentioned that the launch screen should not be an advertisement opportunity and must be identical with the application's first screen which is why the designs above are used.

3. Game news and game news details screen

3.1 iOS (Tabbed Application (i.e. Tab Bar Controller))



Description

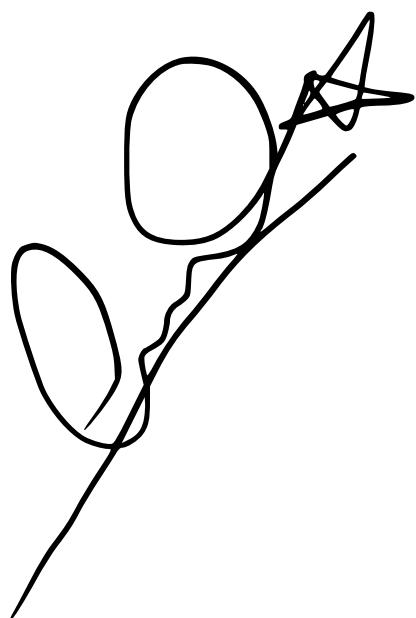
The game news screen (left image) is the application's first screen and will be displayed when the user taps on the "News" tab in the tab bar. It displays multiple game news to the users and provide a brief detail about each news (i.e. time published, news title, source, and image). The user can scroll down to see more news and scroll up to see previous news.

The game news details screen (right image) will be displayed when the user taps on one of the game news. It provides a more comprehensive detail about the news as well as the description of the news. In addition, the user can perform multiple gestures in this screen.

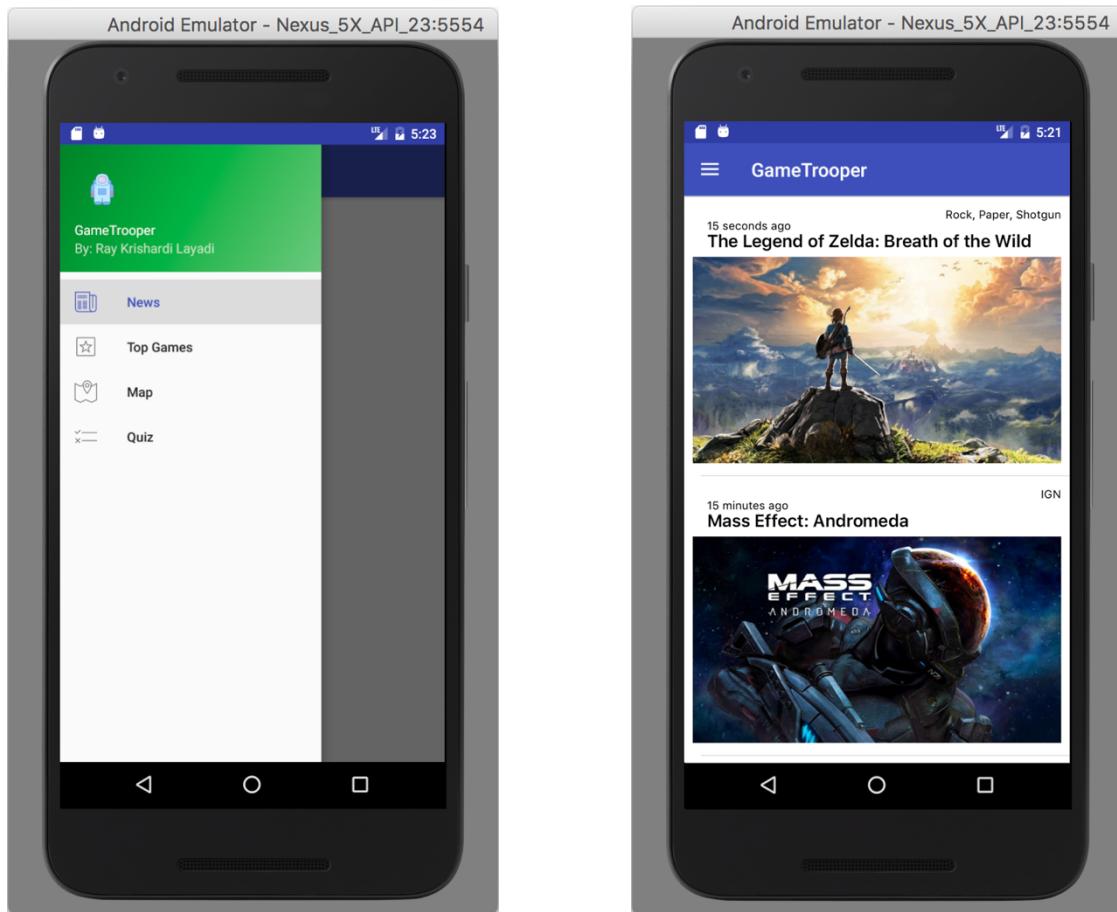
These gestures include:

1. Swipe right to go back to the game news screen (user can also use the “< News” or back button)
2. Double tap for smart zoom
3. Pinch to zoom in and zoom out

In relation to the iOS and Android Design Guidelines, the appropriate built-in font style, size, and color are used to cater for readability of the content.



3.2 Android (Navigation Drawer Activity)



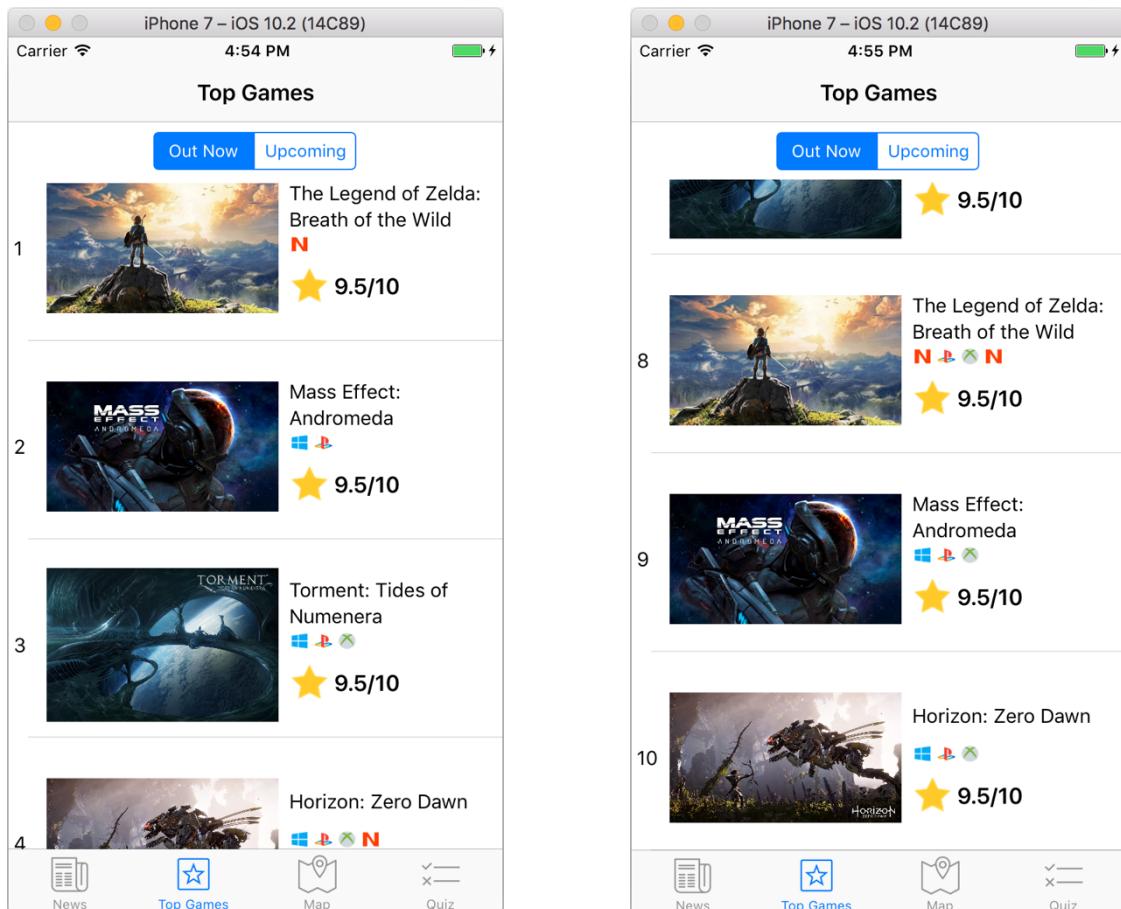
Description

One of the major difference between the iOS and Android design is on the navigation pattern. In this case, iOS uses the tab bar to navigate between news, top games charts, map, and quiz while Android uses the drawer menu (left image). Typography (e.g. font style) between the two platforms will also be different.

The right image is the game news screen in Android. It will behave the same way as the iOS version. In addition, it will also display the same game news details screen when the user taps on one of the game news.

4. Top Games screen

4.1.1 iOS (Top Games “Out Now”)



Description

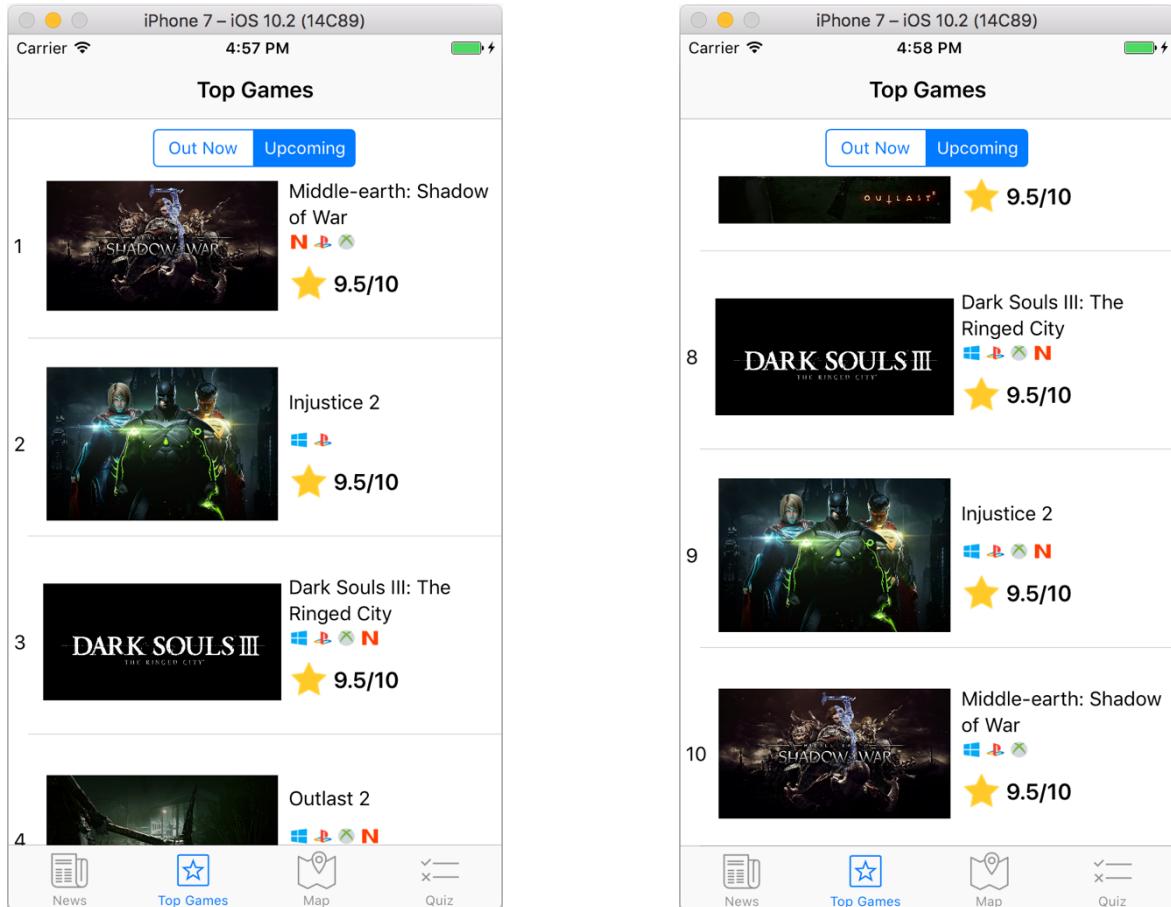
The top games “Out Now” screen will be displayed when the user taps on the “Top Games” tab for the first time or when the user taps on the “Out Now” segment. The user can scroll down and up to see the top 10 games. This screen also provides a brief detail about each game (i.e. game title, available platform, and rating).

In addition, the user can navigate to the top games “Upcoming” screen by tapping the “Upcoming” segment above.

In relation to the iOS and Android Design Guidelines, the appropriate tab in the tab bar and segment in the segmented control are highlighted appropriately so that the users know their current position or section in the application. For example, “Top

“Games” tab and “Out Now” segment are highlighted in these two images to indicate the user’s current position or section in the application.

4.1.2 iOS (Top Games “Upcoming”)



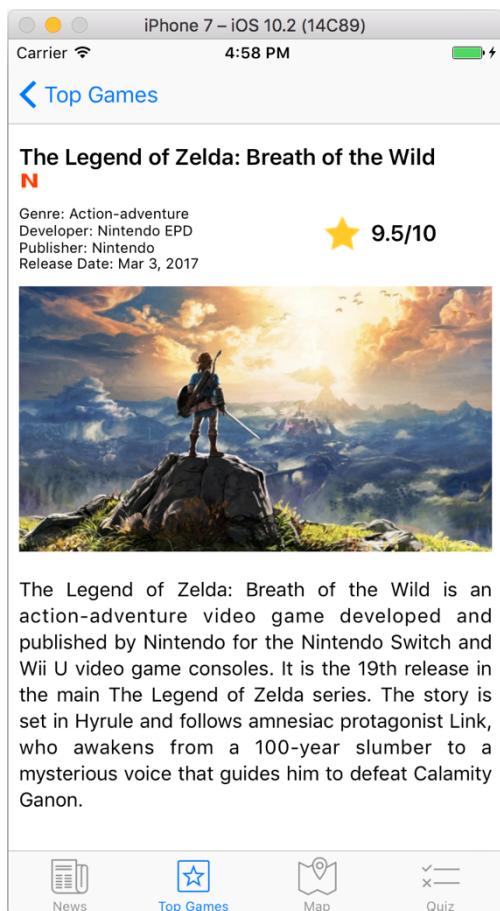
Description

The top games “Upcoming” screen will be displayed when the user taps on the “Upcoming” segment. Just like the top games “Out Now” screen, the user can scroll down and up to see the top 10 games. This screen also provides a brief detail about each game (i.e. game title, available platform, and rating).

In addition, the user can navigate back to the top games “Out Now” screen by tapping the “Out Now” segment above.

Same case with the top games “Out Now” screen regarding the relation to the iOS and Android Design Guidelines.

4.1.3 iOS (Top Games Details)

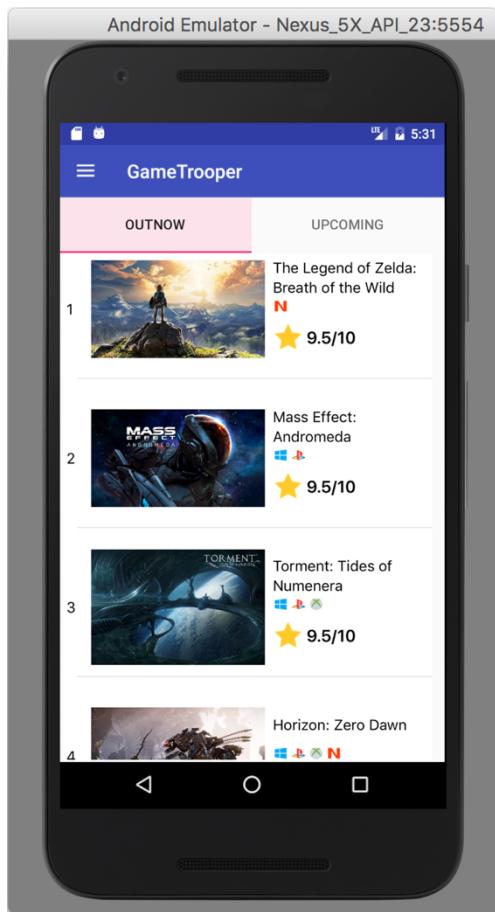


Description

The top games details screen will be displayed when the user taps on a particular game in the top games “Out Now” or ”Upcoming” screen. It provides a more comprehensive detail about the selected game. In addition, just like the game news details screen, the user can also perform the same supported gestures in this screen. However, in this case, when the user swipes to the right, the appropriate top games screen will be displayed (user can also use the “< Top Games” or back button).

This screen has the same case with the game news screen and game news details screen regarding the relation to the iOS and Android Design Guidelines.

4.2 Android (Top Games)



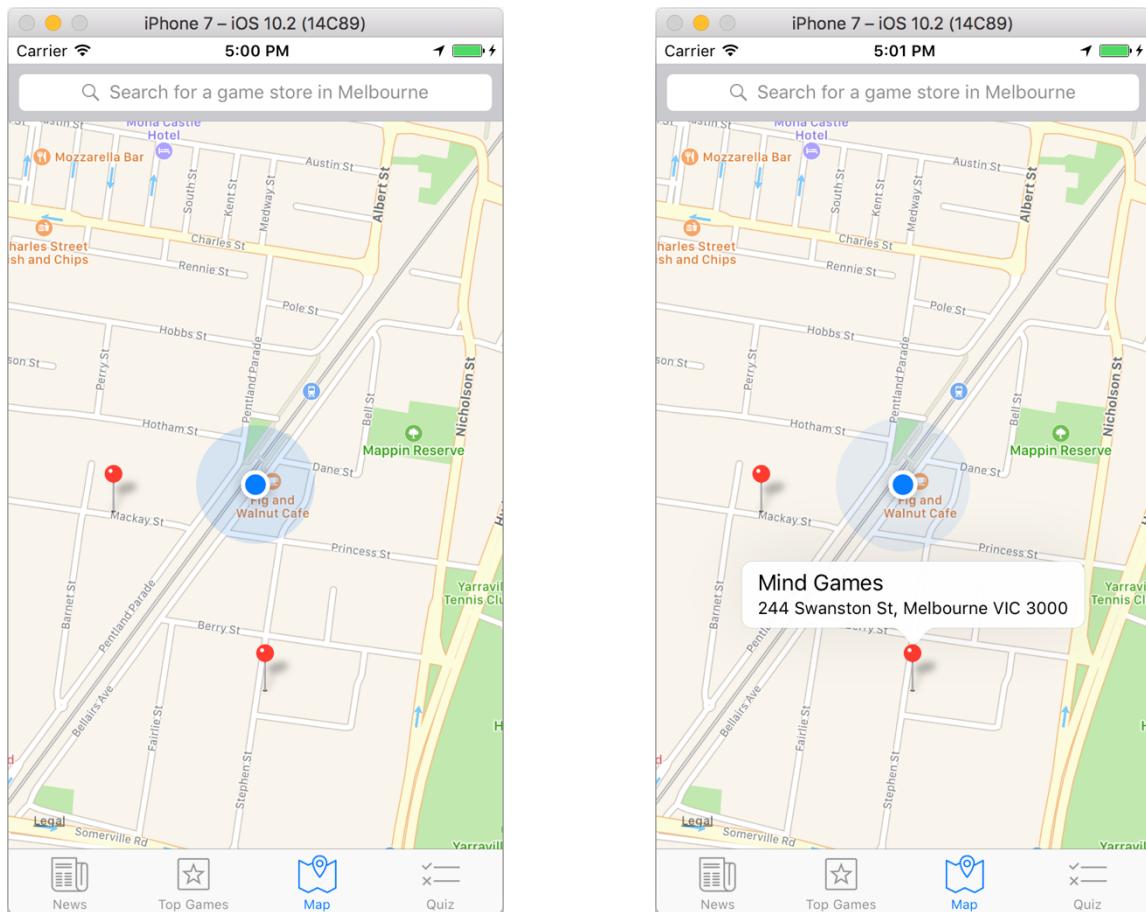
Description

The Android top games screen (“Out Now” and “Upcoming” screens) will behave the same way as the iOS version. In addition, it will also display the same top games details screen as the iOS version when the user taps on one of the game.

The difference between the iOS and Android design for the top games screen is on the segmented control. In this case, iOS uses segmented control to navigate between “Out Now” and “Upcoming” games while Android uses the tab host. Typography (e.g. font style) between the two platforms will also be different.

5. Map screen

5.1 Map main screen



Description

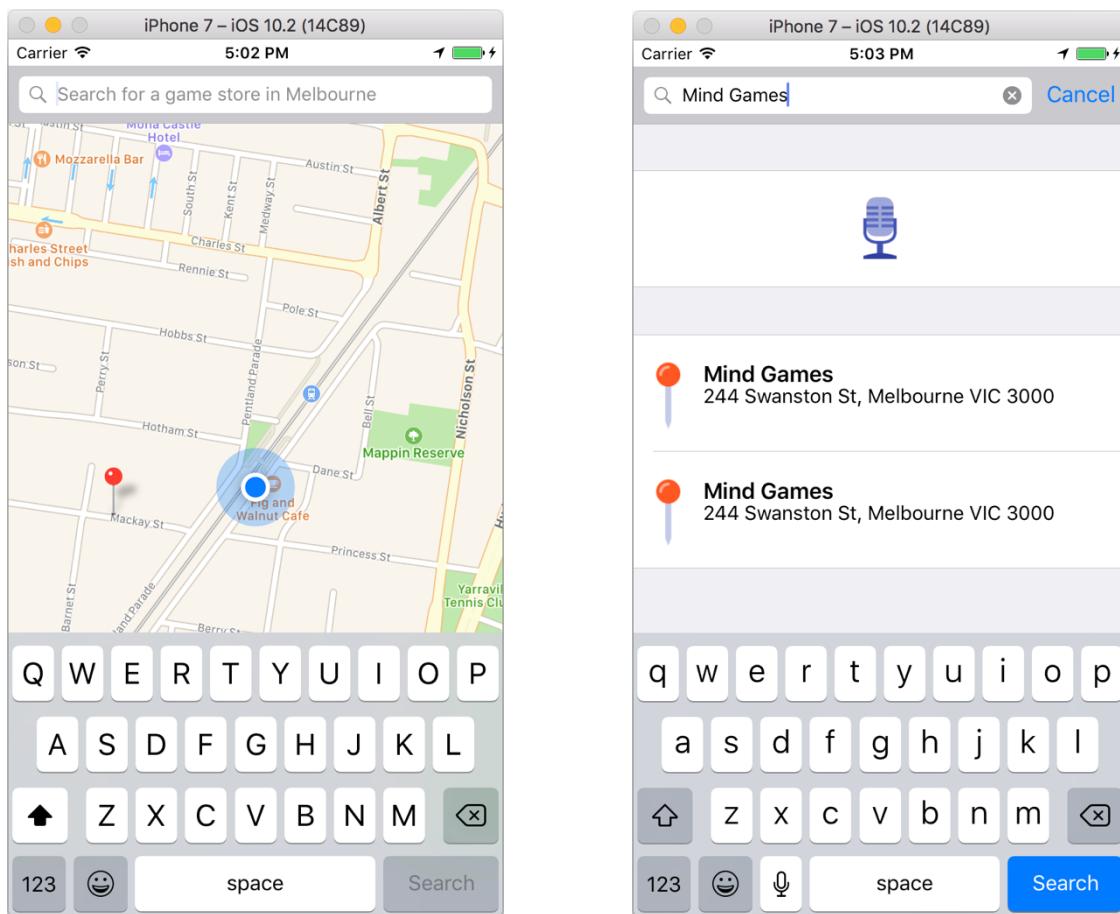
The map main screen will be displayed when the user taps on the “Map” tab in the tab bar. This screen will display annotations (pins) of all local game stores in Melbourne and the user’s current location. It will also automatically update the user’s current location when the user moves to another location.

In addition, when the user taps on one of the game store annotation, the game store name and address will be displayed on the screen (right image).

The difference between the iOS and Android version is on the Maps and Geolocation technology. In this case, iOS uses the Apple Maps API while Android uses Google Maps API.

In relation to the iOS and Android Design Guidelines, the appropriate annotation and user location's symbols are used properly. In addition, the search bar is also positioned at the top with the appropriate placeholder text which mimics other native map application from different platforms. Therefore, users from different platforms are familiar with the design which ultimately provides ease of use.

5.2 Search and search result screen



Description

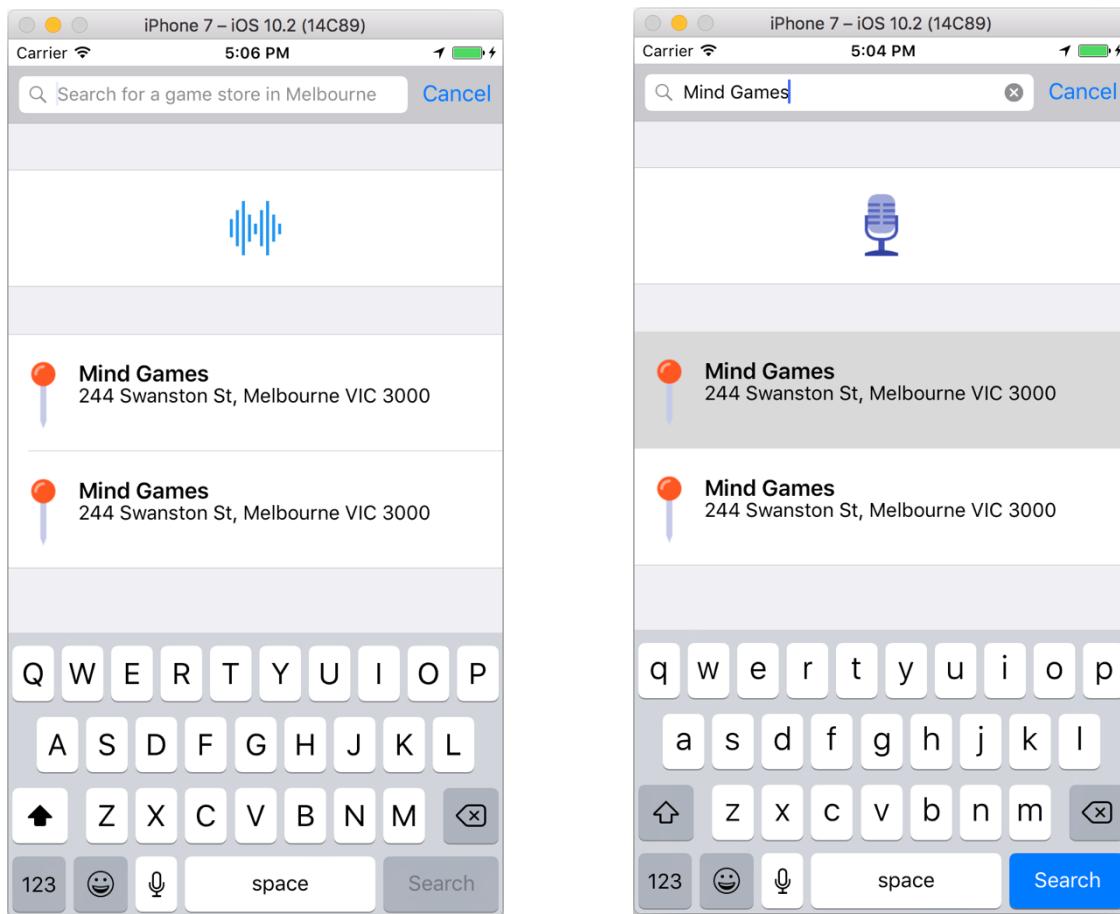
The search screen (keyboard and cursor inside the search bar) will be displayed when the user taps on the search bar at the top of the map view. The user then can search for a game store in Melbourne by typing the desired game store name with the keyboard.

When the user starts typing with the keyboard, the search result screen will be displayed. This screen provides game store name and address that corresponds to the user game store name input. In addition, a microphone button will also be displayed. This microphone button is used to perform speech recognition and find the desired game store name.

In addition, the android implementation will also adopt the same design.

This screen has the same case with the map main screen regarding the relation to the iOS and Android Design Guidelines. However, in this case, a well-known microphone button for speech recognition and annotation pin symbols are used.

5.3 Speech recognition recording and result screen



Description

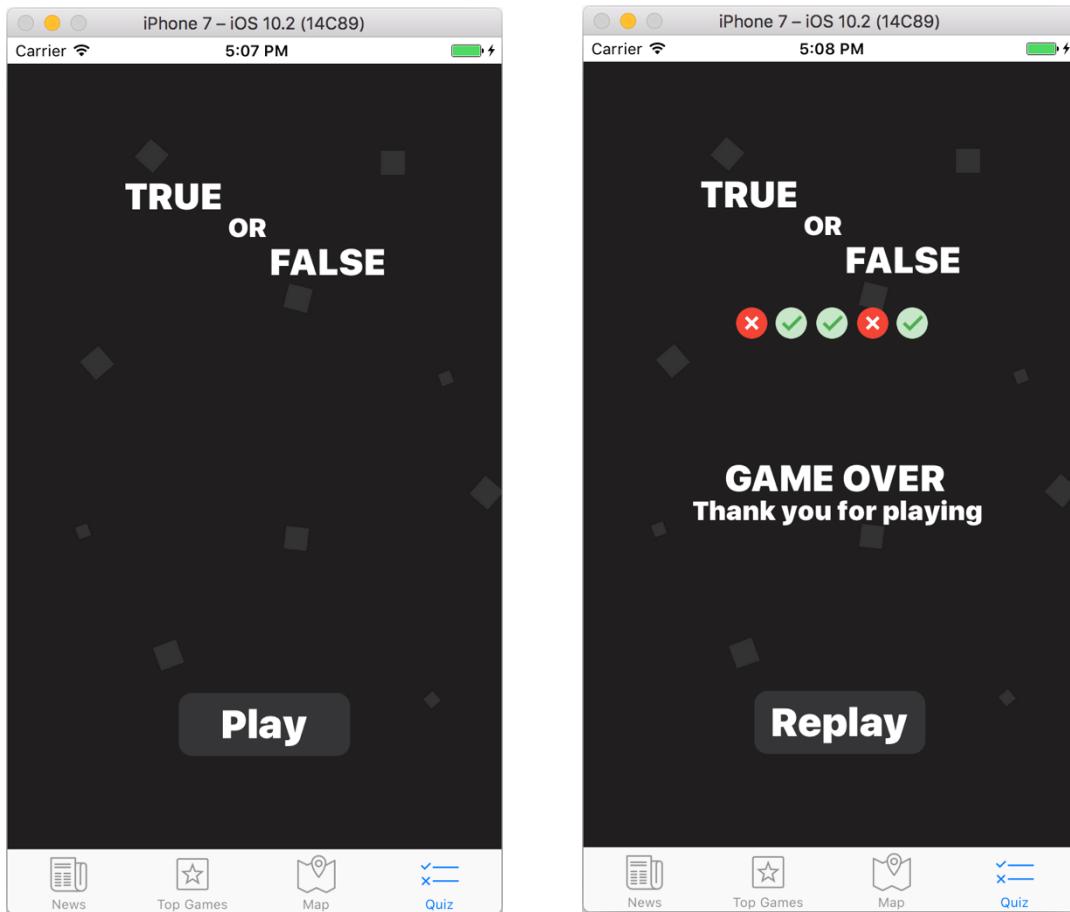
When the user taps on the microphone button, the microphone button will change to an audio wave image (left image) which indicates that the application is currently recording user's audio to perform speech recognition on that recording. After performing speech recognition, the desired game store will be automatically selected (right image) and its corresponding name and address will be displayed in the map view.

In addition, the android implementation will also adopt the same design.

This screen has the same case with the map main screen and search screen regarding the relation to the iOS and Android Design Guidelines. However, in this case, a well-known audio wave image that indicates recording is taking place for speech recognition is used.

6. Quiz screen

6.1 Quiz main screen and Quiz game over screen



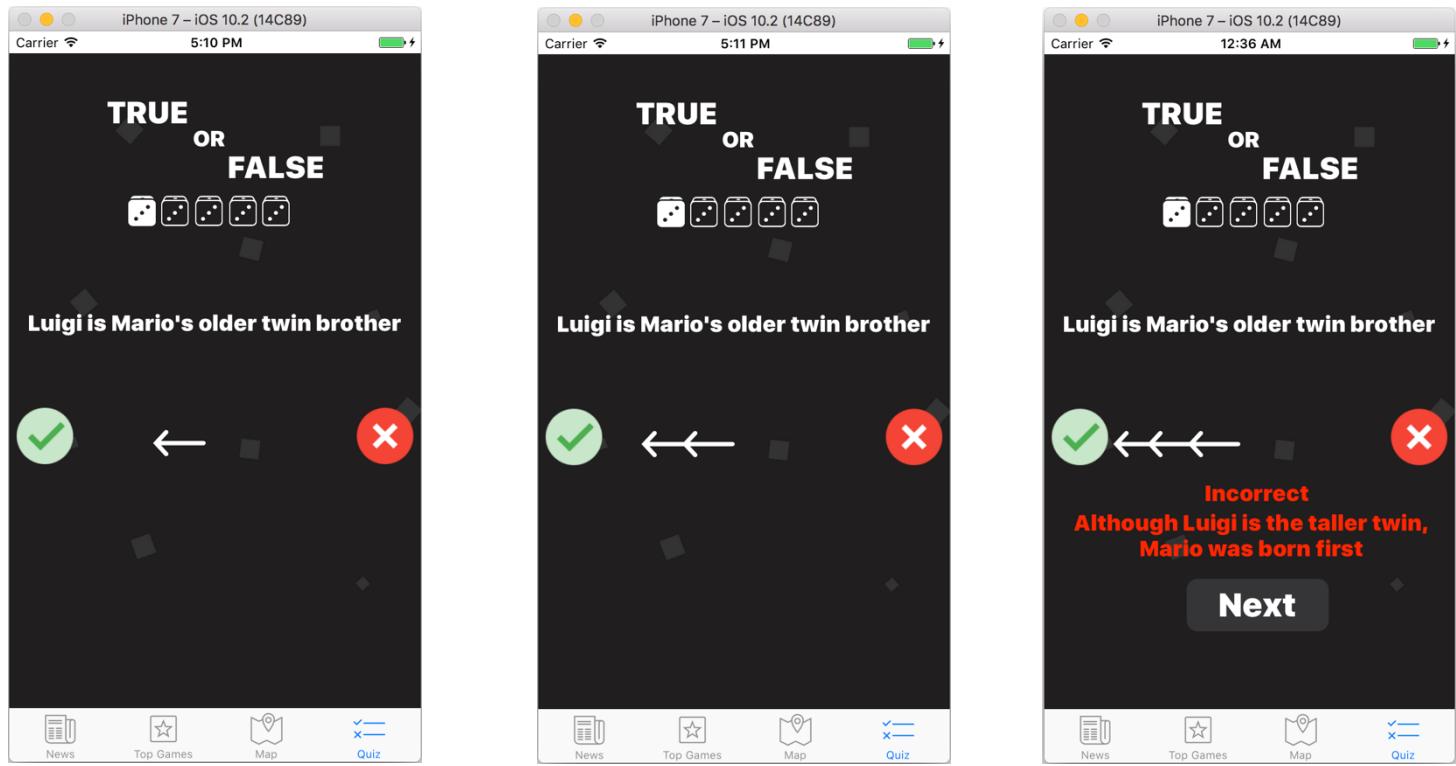
Description

The quiz main screen will be displayed when the user taps on the “Quiz” tab in the tab bar for the first time or when the user taps on the replay button in the quiz game over screen. This screen will display the appropriate quiz game title and play button.

The quiz game over screen will be displayed when the user has answered all the 5 questions. This screen will display the appropriate quiz game title, all correct and incorrect answered questions (represented by the small green check mark and red cross mark), game over title, and replay button.

In relation to the iOS and Android Design Guidelines, the appropriate images and built-in font style, size, and color are used to cater for readability of the content. For example, green check mark image represents true while red cross mark image represents false. Furthermore, the game progress is clearly articulated by highlighting/filling the dice image and answered questions are displayed accordingly with the small green check mark and red cross mark images. Moreover, the use of font color green and red to determine whether the user answered the question correctly or not.

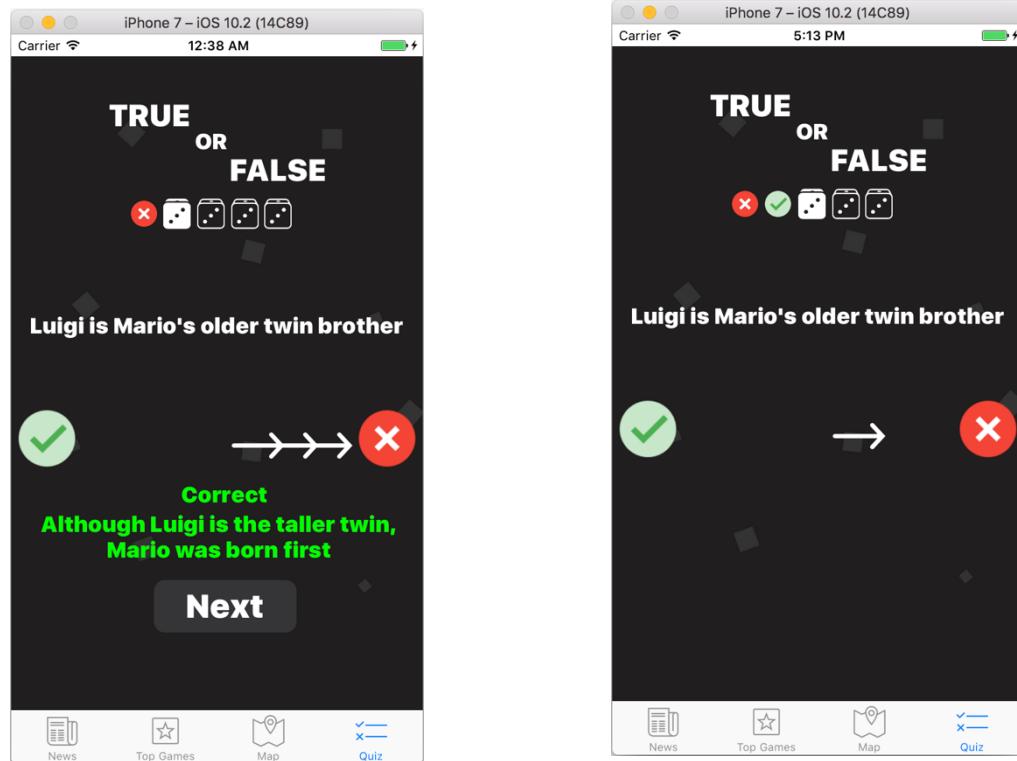
6.2 Quiz in-game screen



Description

The quiz in-game screen will be displayed when the user taps on the “Play” button in the quiz main screen. This screen will display the appropriate quiz game title, game progress (dice, filled dice, small green check mark, and small red cross mark), question, user response mechanism (left and right arrow, big green check mark, and big red cross mark), answer, and next button.

The user moves the left or right arrow by tilting their device to the left or right. The left or right arrow has three stages. The leftmost image (1st image with 1 arrow) displays the first stage, the middle image (2nd image with 2 arrows) displays the second stage, and the rightmost image (3rd image with 3 arrows) displays the third stage. When the arrow reaches the third stage, the next button and the answer to the question will be displayed with the appropriate color (green means the answer is correct while red means the answer is incorrect).



When the user taps on the “Next” button, the game progress will be updated accordingly and new question will be asked. After the user has completed all the questions, the game over screen will be displayed

Same case with the quiz main screen and quiz game over screen regarding the relation to the iOS and Android Design Guidelines.

Scope and Limitations

Mandatory scope

The following list highlights the mandatory features and its priorities:

1. The ability to display game news from variety of game news websites
2. The ability to display more detailed information about the selected game news
3. The ability to display the top 10 current and upcoming games including all relevant information (e.g. available platform, review score, etc.)
4. The ability to display more detailed information about the selected game (e.g. game description, genre, developer, publisher, release date, etc.)
5. The ability to pin local game stores in Melbourne on maps
6. The ability to track user's current location
7. The ability to search for game stores
8. The ability to perform speech recognition and find the desired game stores
9. The ability to play True or False quiz game

*Detailed explanation about each functionality can be found in the **Application Functionality**, **Android and iOS Interface Design Storyboard Mockups**, and **iOS and Android Technology Considerations** sections

These functionalities are crucial for this application because it conforms to the three main objectives that this application is planning to achieve mentioned in the Introduction part of this application design specification.

- The first main objective of this application which is to allow users to stay up-to-date with the current popular new games as well as the new upcoming games is covered by point 1 to 4
- The second main objective of this application which is to assist users in finding local game stores in Melbourne is covered by point 5 to 8
- The third main objective of this application which is to provide entertainment to users by allowing them to play a fun True or False quiz game is covered by point 9

Optional scope

The following list highlights the optional features and its priorities:

1. The ability to notify the user when they are not connected to the internet
2. Maintain an offline database for contents that require internet connection (game news, top games, and game stores name and address) to cater for users that still want to read game news and top games when they do not have an internet connection
3. The ability to support multiple gestures (e.g. swipe, double tap for smart zoom, and pinch to zoom in and zoom out) when the user is reading a particular game news or game details
4. Provide a more intuitive and fluid design that improves user experience
5. Provide a magnetic snapping feeling when the user tilt their device to move the arrow from one stage to another in the True or False quiz game

Limitations

- Limited web development knowledge (need to research on how to utilize website's API)
- The need to research on game news websites that provide free API keys
- The need to research on game news websites API that provide the desired attributes for the application
- The need to redesign game news and top games screens to suit the attributes provided by the game news websites API
- The need to revise knowledge and try to understand how to handle JSON data in iOS and Android
- The need to revise knowledge and try to understand SQLite Database and Core Data
- The complexity of the user response mechanism in True or False quiz game in which there is a need to ensure that it provides a good user experience (i.e. the user must feel good when moving the arrow to the left or right by tilting their device)

- The need to research on the Firebase online database to provide offline support for game news and top games to users
- Assignments and tests from other FIT units

Project Timeline

Week05-06	<p>Research on how to utilize website's API Research game news websites that provide free API keys Research game news websites API that provide the desired attributes for the application Might need to redesign the game news and top games screens to suit the attributes provided by the game news websites API</p>
Week06-08	<p>Need to revise knowledge on how to handle JSON data in iOS and Android Need to revise knowledge on SQLite Database and Core Data Start implementing the game news and top games screen functionalities</p> <p>Games news (game news and game news details screen):</p> <ol style="list-style-type: none"> 1. Able to display list of game news from various sources and include all relevant information (source name, time published, title, and image) <i>*game news screen</i> 2. Able to display detailed information about the selected/tapped news <i>*game news details screen</i> <p>Top games (top games screen):</p> <ol style="list-style-type: none"> 1. Able to display list of top 10 games currently out now <i>*top games "Out Now" screen</i> 2. Able to display list of top 10 upcoming games <i>*top games "Upcoming" screen</i> 3. Able to display detailed information about the selected/tapped game <i>*top games details screen</i>
Milestone 1	
Week09	<p>Further research on the Apple Maps, Google Maps, and Google Places API Further research on the speech recognizer library/framework Start implementing the Map screen functionalities</p>

- Map (map screen):
1. Able to pin local game stores in Melbourne on maps **map main screen*
 2. Able to track user's current location **map main screen*
 3. Able to search for game stores using keyboard **search and search result screen*
 4. Able to perform speech recognition and find the desired game stores **speech recognition recording and result screen*

Milestone 2	
Week10-11 (Prototype 1)	<p>Further research on the sensor and core motion framework</p> <p>Start implementing the Quiz game</p> <p>Quiz game (quiz screen):</p> <ol style="list-style-type: none"> 1. Able to perform all functionalities mentioned in the “Application Functionality” section
Milestone 3	
Week11-12 (Prototype 2)	<p>Continue implementing Quiz game if not finished</p> <p>Implement notification that the users that are not connected to the internet</p> <p>Research and implement Firebase online database to provide offline content support</p> <p>Implement support for multiple gestures (e.g. swipe, double tap for smart zoom, and pinch to zoom in and zoom out) when the user is reading a particular game news or game details</p>
Milestone 4	
Week13-14	<p>Continue implementing notification, Firebase, and gestures if not finished</p> <p>Testing and Debugging</p> <p>Implement other optional functionalities if there is an extra time</p>
Final Application	

*Detailed explanation about each functionality can be found in the **Application Functionality**, **Android and iOS Interface Design Storyboard Mockups**, and **iOS and Android Technology Considerations** sections