A. Software Requirement Specification

A.1 Introduction

The introduction of the Software Requirements Specification (SRS) provides an overview of the entire SRS with purpose, Intended Audience and Reading suggestions and overview of the SRS. The aim of this document is to gather and analyze and give an in-depth insight of the complete **AkiraToeic** by defining the problem statement in detail. Nevertheless, it also concentrates on the capabilities required by stakeholders and their needs while defining high-level product features. The detailed requirements of the **AkiraToeic** are provided in this document.

A.1.1 Purpose

The purpose of this Software Requirements Specification (SRS) section is to provide consistent and complete description of the requirements for **AkiraToeic**. The requirements will be presented using textual descriptions to explain concepts, different types of diagrams to illustrate complicated interactions, and tables to relate relevant information.

A.1.2 Intended Audience and Reading suggestions

This section is intended for all individuals participating in and/or supervising the **AkiraToeic** project. Readers interested in a brief overview of the product should focus on the rest of Part 1 (introduction), as well as Part 2 of the section (Overall Description), which provide a brief overview of each aspect of the project as a whole. These readers may also be interested in part 6 (Key Milestones) which lays out a concise timeline of the project. Readers who wish to explore the features of **AkiraToeic** in more detail should read on to Part 3 (System Features), which expands upon the information laid out in the main overview. Part 4 (Platform Selection) offers further technical details, including information software platforms on which the application will run. Part 5 (System Model) will provide some visual models of the application.

A.1.3 Project scope

The main objective of developing this application is creating a new approach for students

to improve their TOEIC skills. Students could learn English in more fun and active ways.

A.2 Overview

A.1.4 Product perspective

This application aims at providing students with an easy, flexible, joyful way of studying and preparing for TOEIC test. It uses a gamification approach to train students, which means that this application is neither English theoretical training tool nor a brain-free game.
This app will be deployed on HTML firstly and considerably developed on multiple platforms (Windows, Android, iOS) and should be free to download and use.

A.1.5 Product features

This section offers a brief outline and description of the main features and functionalities of the **AkiraToeic**. The features are split into two major categories: core features and additional features. Core features are essential to the application's operation, meanwhile additional features simply add new functionalities. The latter features will only be implemented as time permits.

A.1.5.1 Core features

- USER REGISTRATION & WELCOME
  - Appear when the user access to website's home page
  - Allow the user to new account create using their Facebook or Google + accounts
  - Allow login their account by connecting with Facebook or Google+

- LISTEN GAME
  - Show an audio and an input region.
  - User listen and input the word heard.

- WRITE GAME
  - Show a picture and an input region.
  - User input the word corresponding to the picture.

- CONNECT GAME
  - Show two columns of words.
  - User connect the word from one column to the corresponding work from another column.

- PICTURE GAME
  - Show 3 picture and a word.
  - User choose the picture corresponding to the word.

A.1.5.2 Addition features

- MENU
  - Allow user to choose between different TOEIC levels.
  - Allow user to choose between different skills and lessons.

- USER RANKING TABLE
  - Show the top ranking of users.

- USER INFORMATION MANAGEMENT
  - Show user name and account's picture.
  - Show the levels and the experience of the user.

A.1.6 User classes and characteristics

The users of the AkiraToeic Game are mainly college students, TOEIC test takers and teachers of English.

A.1.7 Operating environment

The main component of the **AkiraToeic** project is the web platform, which will be deployed using PHP and JavaScript with HTML/CSS for designing. The application is not resource- or graphics-intensive, so there are no practical hardware constraints. The app will rely on several functionalities built into Angular JS and Bootstrap 3 framework, so ensuring appropriate usage of the API will be a major concern.

A.1.8 Design and implementation constraints

The primary design constraint is the web platform. Creating a user interface which is both effective and easily navigable will pose a difficult challenge. Other constraints such as limited memory and processing power are also worth considering. **AkiraToeic** is meant to be quick and responsive, even when dealing with large groups and transactions, so each feature must be designed and implemented with efficiency in mind.

A.1.9 User documentation

The primary goal of **AkiraToeic** is to help user approach with TOEIC. Consequently, the application will be designed to be as simple to use as possible. Nonetheless, users may still require some supplementary information about the game. The application will contain a Help menu.
The Help menu is a collection of instructions of each game's rule and how to play them. It will be covered by both English and Vietnamese.

A.3 Specific Requirements

A.1.10 User Requirements

The application has to categorize users and lessons into different levels (TOEIC 350, TOEIC 500, TOEIC 650, and TOEIC 800)
Application must help users to improve their English skills: Listening, Vocabulary, Writing specifically for TOEIC test.
Application should have account management function and users can use their Facebook/Google account to log in.

A.1.11 System Requirements

A.1.11.1 Functional Requirements

No	Functional Requirements	
1.	There must be a menu for user to select between different TOEIC levels	Application Overall
2.	There should be a region within the website in which displays user information such as: Username, day of use, current levels	
3.	There should have a ranking board which show the rank of users	
4.	When each game is finished, the application must report to users their score, experiences, etc.	
5.	User can use both mouse and keyboard to interactive with games.	
6.	Each game initially has 3 life and a progress bar display user's current progress.	
7.	Each question must include **a picture, an audio and an input region** for user to enter their name	Write Game
8.	When user presses **Enter** key, application must check if user's answer is correct, display the correct answers and play corresponding sound.	

9.	Each question must include **a word, 3 pictures** for user to select.	Listen Game
10.	When user presses **Enter** key, application must check if user's answer is correct, display the correct answers and play corresponding sound.	
11.	There are two columns of word: English (in blue color) and Vietnamese (in red).	Connect Game
12.	When user click to a pair of words, the application must check if user's answer is correct, display the correct answers and play corresponding sound.	
13.	Only 1 item in each column could be chosen at a time.	
14.	Each question must include **an audio, a slow-speed audio, and an input region** for user to enter their answers.	Listen Game
15.	When user presses **Enter** key, application must check if user's answer is correct, display the correct answers and play corresponding sound.	
16.	When user submit their answer (i.e., press Enter key), the application must not allow user to change their answer.	

A.1.11.2 Non - Functional Requirements

A.1.11.2.1 Usability

- Graphical User Interface
  - The interface has to be lightweight, easy to use and well-styled.
  - The system shall provide an image for each game in the list.
  - The system shall provide use of icons and toolbars.

B. Accessibility
  - The system use English language and could conclude Vietnamese instruction.

A.1.11.2.2 Reliability & Availability

- The answer and question in the game have to be correct.
- The system has to be active whenever a failure happens.

A.1.11.2.3 Performance

- The application has to run smoothly and stable.

A.1.11.2.4 Data Storage

- The data of questions and answers are stored as JSON file type.

B.1 Scenarios

A.1.12 Account Login

Description: When user accesses to website, the login interface will appear. There are two options for user: Facebook and Google. User chooses his way to login by clicking on the button. After that, the website will automatically connect with his Facebook or Google account if his account is already been in the browser. If not, a small window will appear for user to login to his account.

A.1.13 Write Game

Description: There will be a picture and an input region. User will have to input the correct word corresponding to the picture. After user inputs the answer, the "kiểm tra" button will light up and ready for user to click. User clicks this button to check whether his answer correct or not. The correct answer will be appear. If the user's answer is correct, the progress bar will increase and user moves to the next question. If the user's answer is incorrect, user will lose a heart (life) and moves to the next question. If user loses 3 hearts, the game will be finished and the result will appear.
If user do not know the picture's word, he could click on the audio button to hear the word.

A.1.14 Picture Game

Description: There will be 3 picture and a word. User will have to choose the picture corresponding to the word by clicking on the picture. The picture clicked will be light up and so the "kiểm tra" button. User clicks this button to check whether his answer correct or not. The correct answer will be appear. If the user's answer is correct, the progress bar will increase and user moves to the next question. If the user's answer is incorrect, user will lose a heart (life) and moves to the next question. If user loses 3 hearts, the game will be finished and the result will appear.

A.1.15 Connect Game

Description: There will be two columns of word: English (in blue color) and Vietnamese (in red). User will have to connect the word from one column to the corresponding work from another column. A word could only connect to one word from the opposite column. User connects word by clicking one word in the right column and after that clicking one word from the left column. If the user's answer is correct, the progress bar will increase and user continues to connect. If the user's answer is incorrect, user will lose a heart (life) and user continues to connect. If user loses 3 hearts, the game will be finished and the

result will appear.

A.1.16 Listen Game

Description: There will be two audio buttons and an input region. User will clicking the audio button to listen the word and input the word heard. After user inputs the answer, the "kiểm tra" button will light up and ready for user to click. User clicks this button to check whether his answer correct or not. The correct answer will be appear. If the user's answer is correct, the progress bar will increase and user moves to the next question. If the user's answer is incorrect, user will lose a heart (life) and moves to the next question. If user loses 3 hearts, the game will be finished and the result will appear.
If the word is spoke too fast, user could click on the "turtle" button to listen to a slow – speaking version of the word.

B.2 Platform selection

A.1.17 Application Development (C#, C++, Java….)

The application is developed using JavaScript mainly.

The user interface will be designed using HTML5/CSS.

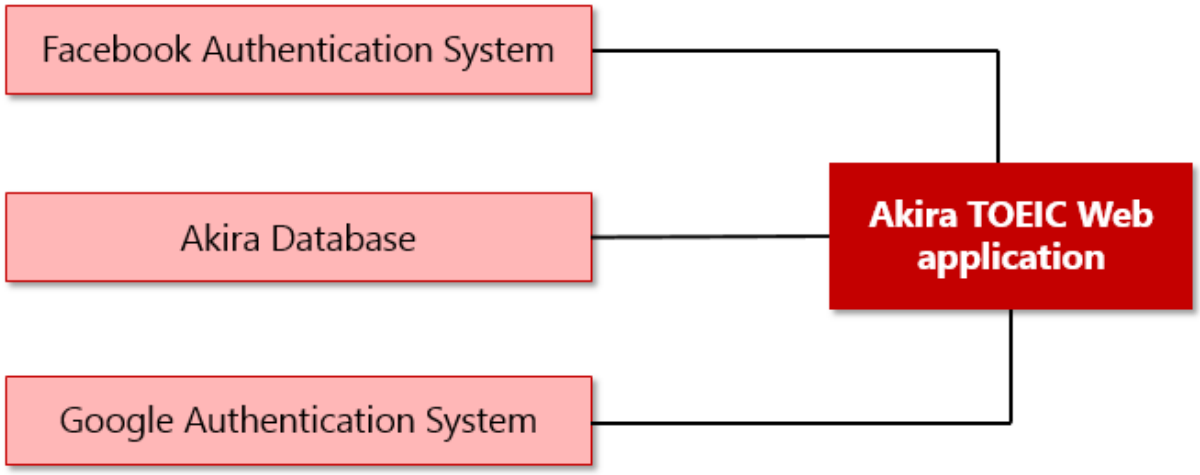
Server site programming language PHP.

A.1.18 Technological Solution

No	Name	Information	Assessment
1	*Angular JS*	*https://angularjs.org/*	*AngularJS, commonly referred to as Angular, is an open-source web application framework maintained by Google and a community of individual developers and corporations to address many of the challenges encountered in developing single-page applications. Its goal is to simplify both development and testing of such applications by providing a framework for client-side model–view–controller (MVC) architecture, along with components commonly used in rich internet applications.*
2	*Bootstrap 3*	*http://getbootstrap.com/*	*Bootstrap is the most popular HTML, CSS, and JS framework for developing responsive, mobile-first projects on the web.* *Bootstrap helps you kickstart the development of webapps and websites.*

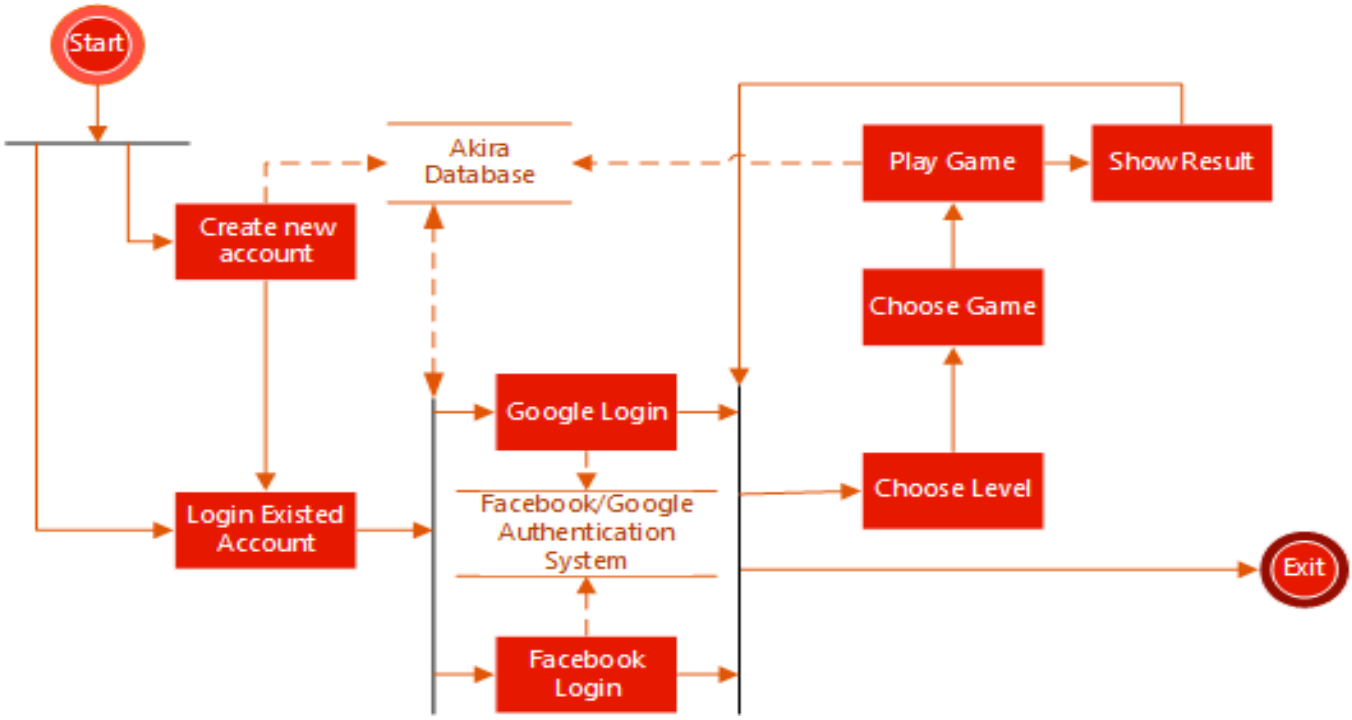
B.3 System Model

A.1.19 Context Model

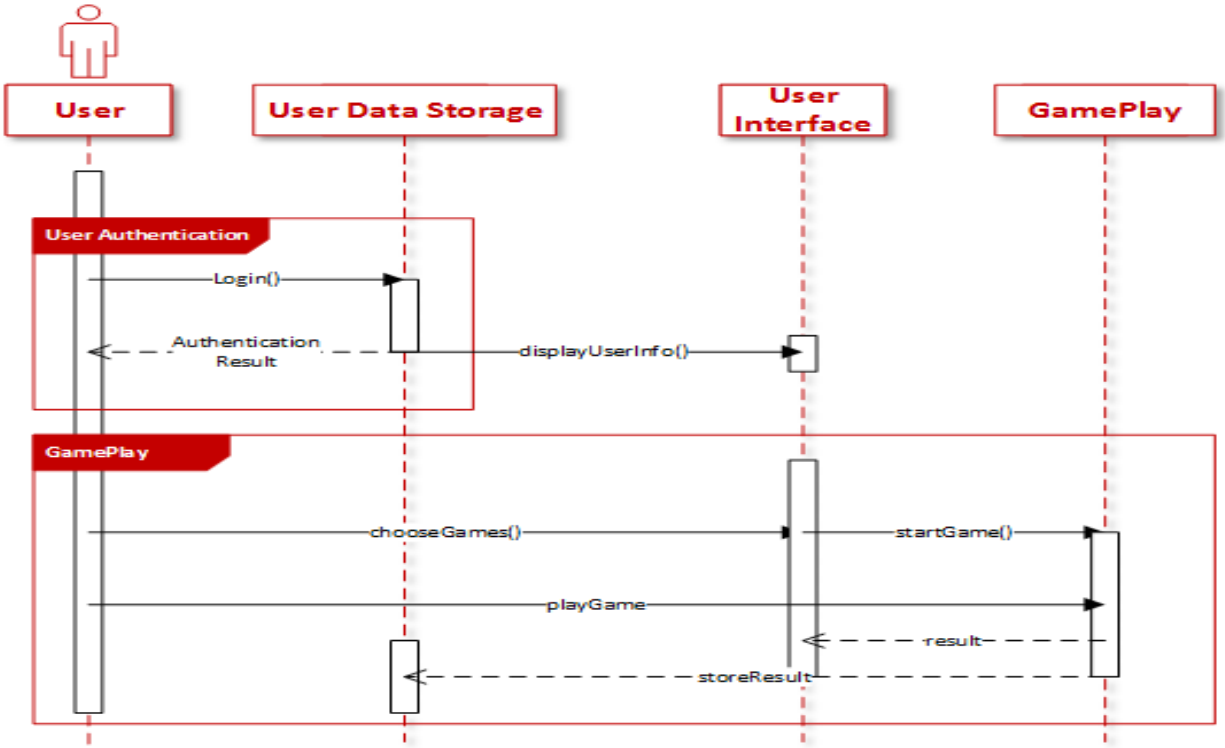


The figure above illustrates the structure of the information system that includes an **AkiraTOEIC** web application. The system connects with Akira Database to archive the information of user and the database for game play. The system also connects to the Facebook/Google Authentication System for user login.

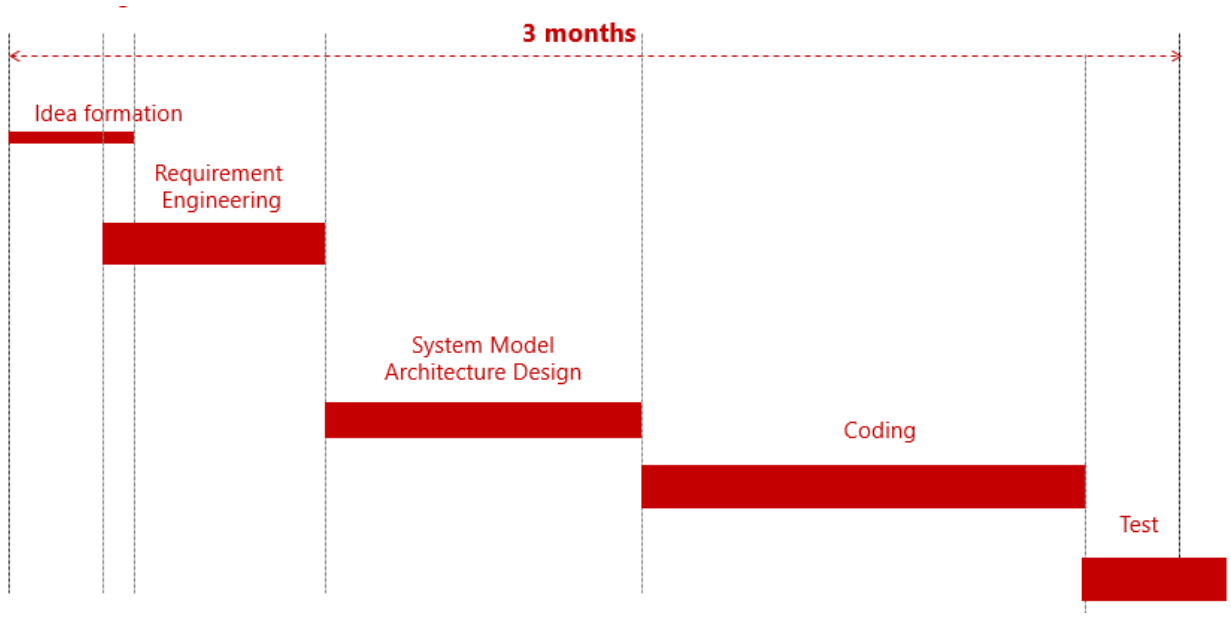
A.1.20 Process Model



A.1.21 Sequence Model



B.4 Key milestones



C. Software Architecture Design

C.1 Introduction

The introduction of the Software Architecture Design (SAD) section provides an overview of the entire SAD with purpose, Intended Audience and Reading suggestions and specification of the software model and architecture. The aim of this document is to gather and analyze and give an in-depth insight of the complete **AkiraToeic** project by showing the model and the architecture of the software.

A.1.22 Purpose

The purpose of this Software Architecture Design (SAD) section is to a comprehensive architectural overview of the system, using a number of different architectural views to depict different aspects of the system. It is intended to capture and convey the significant architectural decisions which have been made on the system.

A.1.23 Intended Audience and Reading suggestions

This section is intended for all individuals participating in and/or supervising the **AkiraToeic** project. Readers interested in a brief overview of the product should focus on the rest of Part 1 (introduction), as well as Part 2 of the section (Architecture Goals and Characteristics), which provide an overall characteristics of the project.

Readers who wish to know the architecture of **AkiraToeic** in more detail should read on to Part 3 (Architecture Views) and Part 4 (Architecture Design), which expands upon the information laid out in the main overview.

A.1.24 Scope

This Software Architecture section provides an architectural overview of the **AkiraToeic** System. The **AkiraToeic** System is being developed by Group 3 to create a new Toeic learning environment.

C.2 Architecture Goals and Characteristics

A.1.25 Security

Reader rights will be grated to any user accessing the application-landing page.

User's information will be protected, only Administrator user can add or remove other Creators and perform other administrative tasks.

A.1.26 Persistence

Data persistence will be addressed using a JSON database.

A.1.27 Reliability/Availability

Reliability/Availability will be addressed through the server platform: www.akira.edu.vn/learn/login.html / www.local.akira.com.

A.1.28 Performance

There is no particular constrains related to system performance.
It is anticipated that the system should respond to any request well under standard database and web server script timeouts (20 seconds), In addition, upload / download times can depend on data size which in turn depends on user input. Therefore, actual performance can be determined only after system deployment and testing.

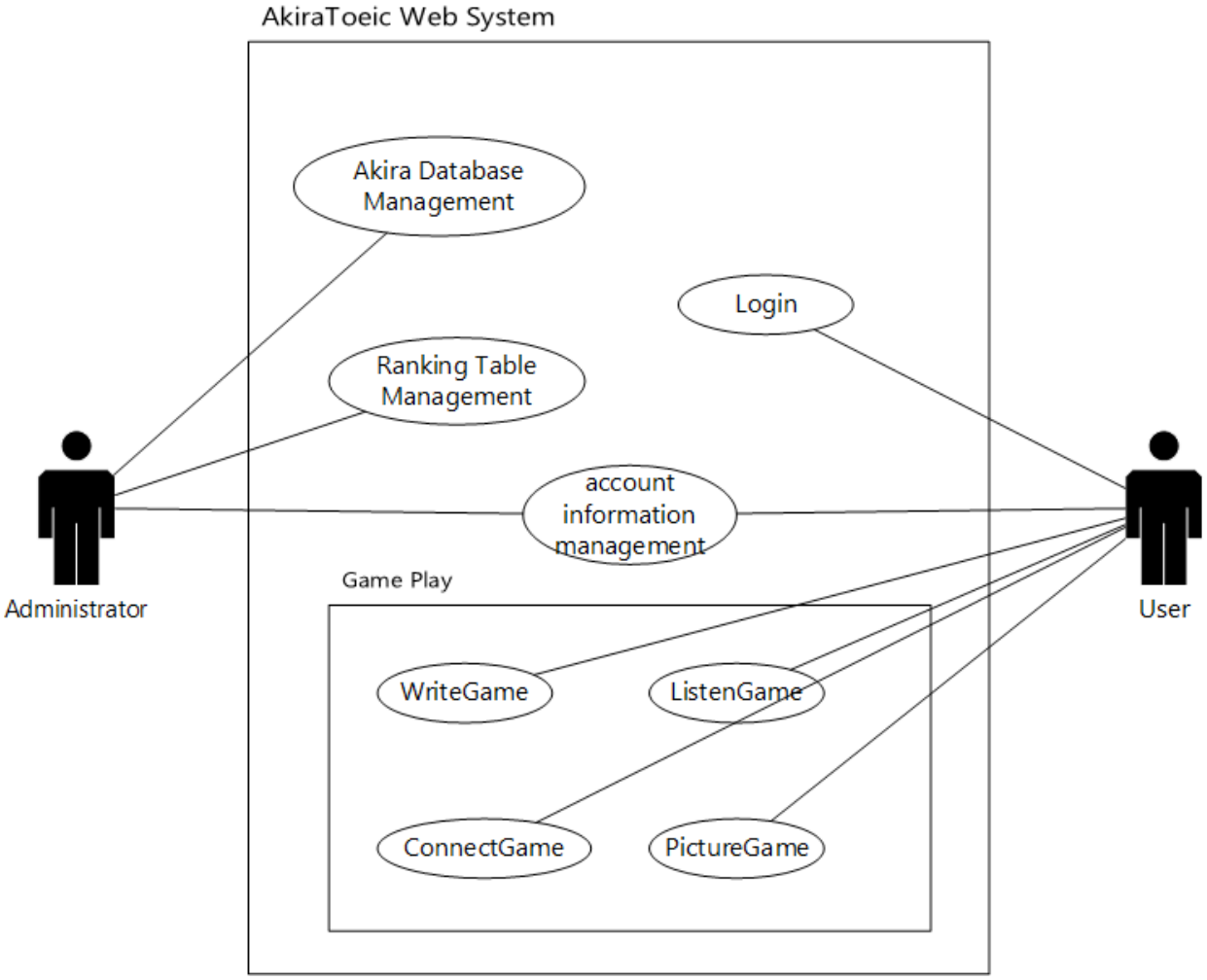
C.3 System Architecture Views

A.1.29 Use-case view

This is a list of use-cases that represent major functionality of the final system [SRS]:

- Akira database management
- Ranking table management
- User Login
- Account information management
- Write Game
- Listen Game
- Connect Game

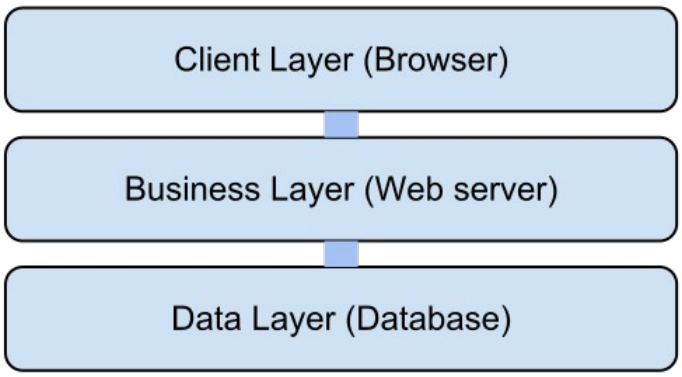
- Picture Game



As described in the actors' correspondence diagram below, web user could be one of three types:

- **Administrator** has enhanced privileges to view, delete, add and handle Akira Database
- **User** could register their account and use the service which are the TOEIC game plays

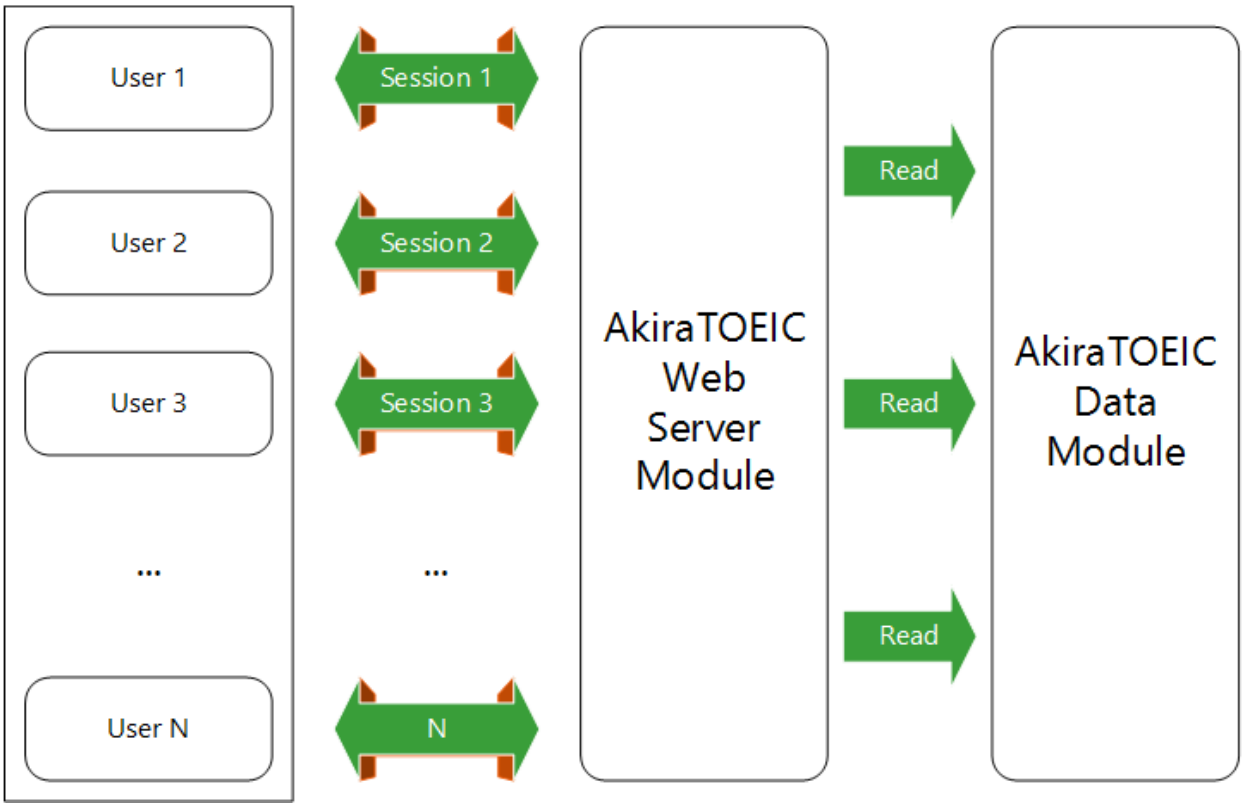
A.1.30 Logical view



The layering model of the **AkiraToeic** application is based on a responsibility layering strategy that associates each layer with a particular responsibility.

This strategy has been chosen because it isolates various system responsibilities from one another, so that it improves both system development and maintenance.

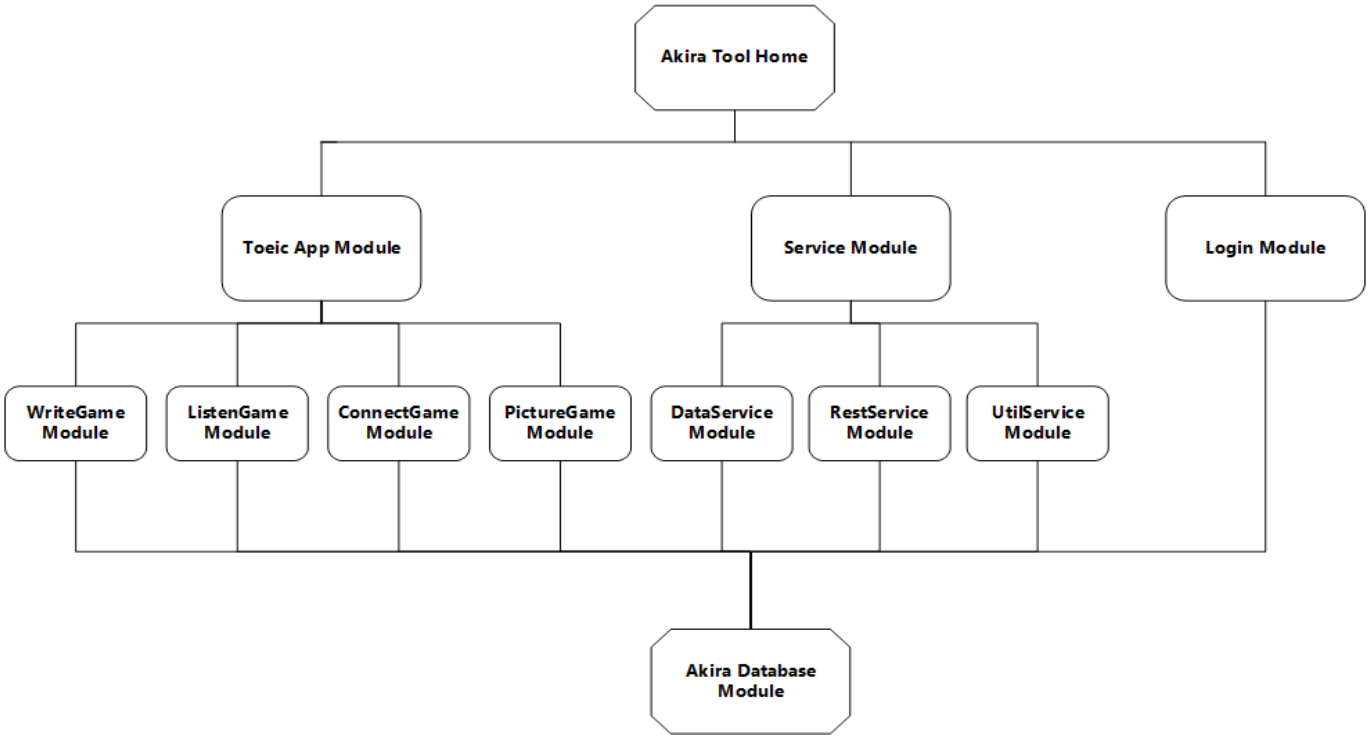
A.1.31 Process view



Due to disconnected nature of HTTP request / response, **AkiraToeic** will handle multiple users simultaneously. Therefore, concurrency issues such as synchronous versus asynchronous mechanisms will be not considered in this document.

A.1.32 Module Decomposition View

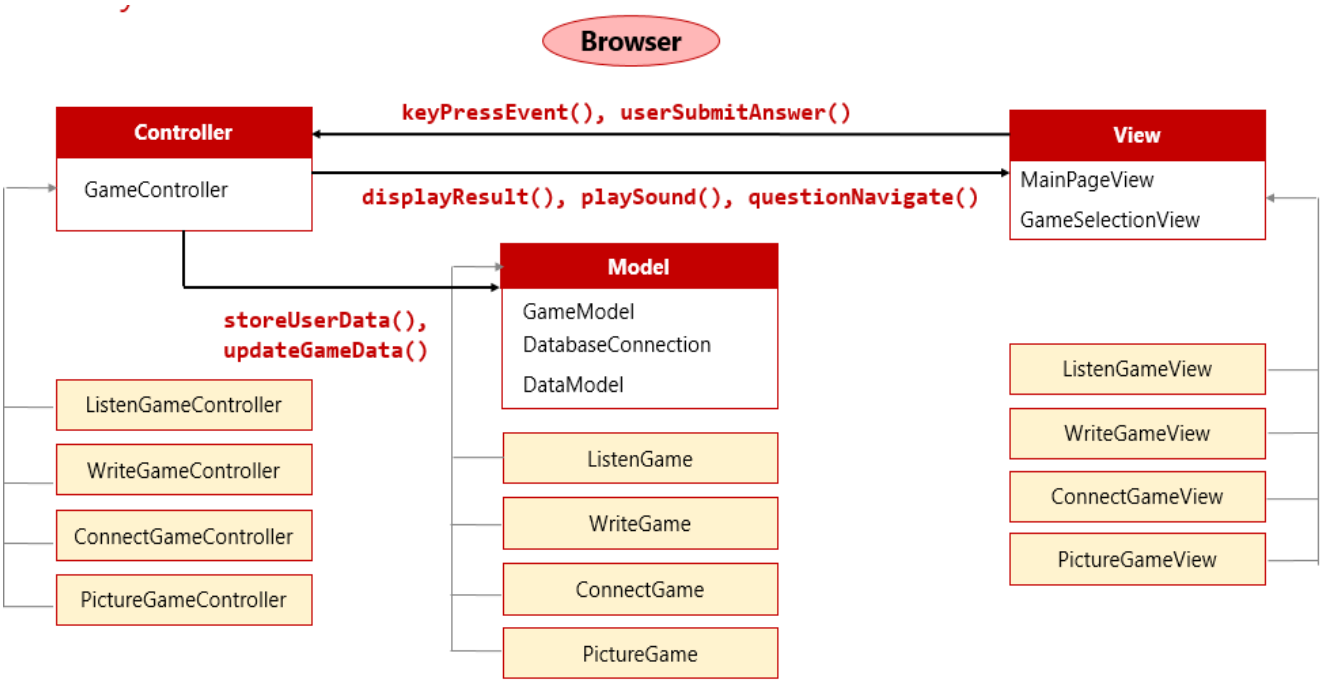
Module decomposition view based on principles separation of concerns and abstraction and supports goals of modifiability and usability.



C.4 System Architecture Design

A.1.33 Architecture Pattern

The **AkiraToeic** project is designed using MVC pattern.



A.1.34 Architecture Specification

- **GameController:** main controller of the application, handle key event listener and navigate to sub module.

toeicCtrls.controller('mainCtrl', function($rootScope, $scope, toeicStar, $routeParams, menuFactory, utilService))

- ListenGameController: controller of the Listen Game
- WriteGameController: controller of the Write Game
- ConnectGameController: controller of the Connect Game
- PictureGameController: controller of the Picture Game

- **Data Object Template**:
  - Grammar object:

```
{
        "id": 1,
        "filename": "1.1.1",
        "english": "We_ask the committee",
        "correct": "shall",
        "wrong": "can",
        "vietnamese": "Chúng tôi sẽ yêu cầu ủy ban "
 }
```

  - Vocab object:

```
{
        "topic": 1,
        "sub": 1,
        "id": 1,
        "filename": "hat",
        "word": "hat",
        "phonetic": "hæt",
        "part of speech": "N",
        "meaning": "mũ ",
        "sentencee": "This hat is cheap at 1000 yen.",
        "sentencev": "Cái nón này giá 1000 yên là rẻ."
   }
```

D. Software Testing Plan

D.1 Introduction

The **AkiraToeic** Software Testing Plan (STP) document provides necessary information required to effectively define the approach to be used in the testing of the project's product. This document is created during the Planning Phase of the project. Its intended audience is the project manager, project team, and testing team. Some portions of this document may on occasion be shared with the client/user and other stakeholder whose input/approval into the testing process is needed.

A.1.35 Objectives

AkiraToeic is a web-based learning application used to create a new environment for students to learn TOEIC in a fun and joyful way. The test team is responsible for testing the product and ensuring it meets their needs. The test team is both the customer and the developer in this project.

A.1.36 Intended Audience and Reading suggestions

This document is intended for all individuals participating in and/or supervising the **AkiraToeic** project. Readers interested in a brief overview of the product should focus on the rest of Part 1 (introduction), as well as Part 2 of the document (Test Plan Characteristics), which provide an overall characteristics of the test plan.

Readers who wish to know the architecture of **AkiraToeic** in more detail should read on to Part 3 (Development Testing Plan), which expands the specification of the testing plan.

D.2 Test Plan Characteristics

A.1.37 Test risks/ issues

The following risks have been identified and the appropriate action identified to mitigate

their impact on the project. The impact (or severity) of the risk is based on how the project would be affected if the risk was triggered. The trigger is what milestone or event would cause the risk to become an issue to be dealt with.

#	Risk	Impact	Trigger	Mitigation Plan
1	Internet Disconnection	High	User could not connect to server	The application load the data from server before starting so user still could use some features of the application.
2	Too many user's request	High	Denial of service	Until now, the application does not have an effectively plan for this.

A.1.38 Items to be tested/ be tested

- Items to be tested:
  - User authentication module
  - Menu graphical
  - Write Game
  - Listen Game
  - Connect Game
  - Picture Game
- Items not to be tested:
  - Ranking table management
  - Feedback feature

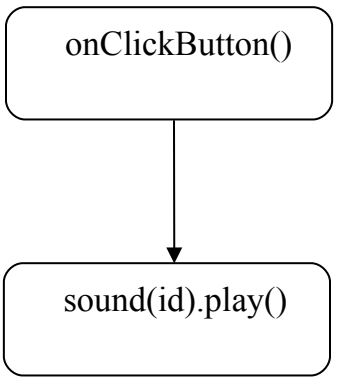
A.1.39 Tests Environments

The application will be tested in the Window 8 environment using Google Chrome version 39.0.2171.95 m

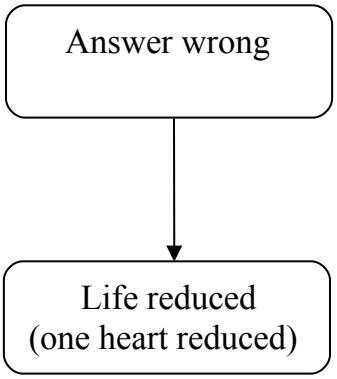
D.3 Development Testing Plan

A.1.40 Unit Testing

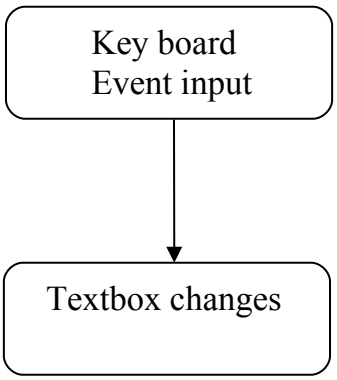
- Play sound : $scope.playSound = function(id, isNormal)



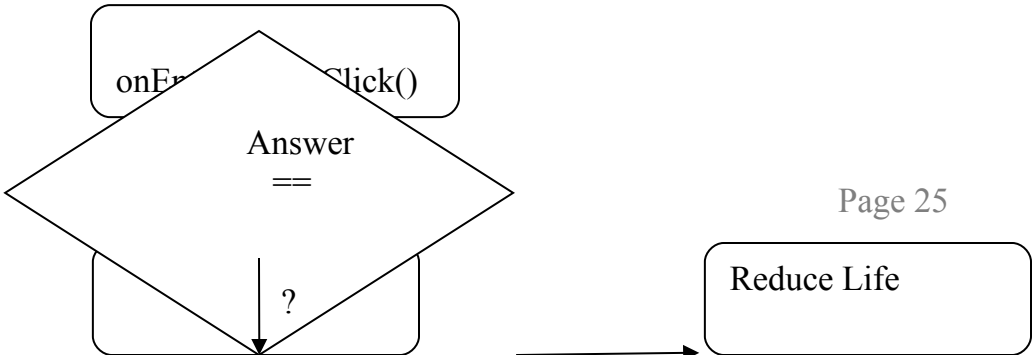
- Remove life : $scope.removeLife = function()



- Key press : $scope.keyPress = function(e, keyCode)



- Check answer : $scope.check = function()



A.1.41 Component Testing

A.1.41.1 Login Module

Story Card: When user accesses to website, the login interface will appear. There are

two options for user: Facebook and Google. User chooses his way to login by clicking on the button. After that, the website will automatically connect with his Facebook or Google account if his account is already been in the browser. If not, a small window will appear for user to login to his account.

Test Case:

- **Input:** User clicks on the button (Facebook/Google)
- **Test:** Application checks from the browse for existed user Facebook/Google accounts. After that, connect to Facebook/Google Authentication System to login.
- **Output:** Continue to next page if there is an account
  Show a small window to choose accounts if there are multiple account
  Show a small window for input user name/password if there is not any account.

A.1.41.2 Write Game

Story Card: There will be a picture and an input region. User will have to input the correct word corresponding to the picture. After user inputs the answer, the "kiểm tra" button will light up and ready for user to click. User clicks this button to check whether his answer correct or not. The correct answer will be appear. If the user's answer is correct, the progress bar will increase and user moves to the next question. If the user's answer is incorrect, user will lose a heart (life) and moves to the next question. If user loses 3 hearts, the game will be finished and the result will appear.
If user do not know the picture's word, he could click on the audio button to hear the word.

Test Case:

- **Input:** String inputted at input region.
- **Test:** Write game module compare the string with the correct answer.
- **Output:** Show the correct answer
  A heart is reduced if the answer is wrong. After that, show a next question.
  Show a next question if the answer is correct

A.1.41.3 Picture Game

Story Card: There will be 3 picture and a word. User will have to choose the picture corresponding to the word by clicking on the picture. The picture clicked will be light up and so the "kiểm tra" button. User clicks this button to check whether his answer correct

or not. The correct answer will be appear. If the user's answer is correct, the progress bar will increase and user moves to the next question. If the user's answer is incorrect, user will lose a heart (life) and moves to the next question. If user loses 3 hearts, the game will be finished and the result will appear.

Test Case:

- **Input:** picture onClickListener().
- **Test:** Picture game module compare the id of the picture with the correct picture's id.
- **Output:** Show the correct answer
  A heart is reduced if the answer is wrong. After that, show a next question.
  Show a next question if the answer is correct

A.1.41.4 Connect Game

Story Card: There will be two columns of word: English (in blue color) and Vietnamese (in red). User will have to connect the word from one column to the corresponding work from another column. A word could only connect to one word from the opposite column. User connects word by clicking one word in the right column and after that clicking one word from the left column. If the user's answer is correct, the progress bar will increase and user continues to connect. If the user's answer is incorrect, user will lose a heart (life) and user continues to connect. If user loses 3 hearts, the game will be finished and the result will appear.

Test Case:

- **Input:** word onClickListener
- **Test:** After two onClickListener events, compare the word from 2 last buttons.
- **Output:** A heart is reduced if the answer is wrong.
  Disable two last clicked button if the answer is correct.

A.1.41.5 Listen Game

Story Card: There will be two audio buttons and an input region. User will clicking the audio button to listen the word and input the word heard. After user inputs the answer, the

"kiểm tra" button will light up and ready for user to click. User clicks this button to check whether his answer correct or not. The correct answer will be appear. If the user's answer is correct, the progress bar will increase and user moves to the next question. If the user's answer is incorrect, user will lose a heart (life) and moves to the next question. If user loses 3 hearts, the game will be finished and the result will appear.
If the word is spoke too fast, user could click on the "turtle" button to listen to a slow – speaking version of the word.

Test Case:

- **Input:** String inputted at input region.
- **Test:** Write game module compare the string with the correct answer.
- **Output:** Show the correct answer
  A heart is reduced if the answer is wrong. After that, show a next question.
  Show a next question if the answer is correct

A.1.42 System Testing

Testing plan: