



CALIFORNIA STATE UNIVERSITY  
**FULLERTON™**

Department of Computer Science

This project has been satisfactorily demonstrated and is of suitable form.

This project report is acceptable in partial completion of the requirements for the Master of Science degree in Software Engineering.

EZ-Learning Web Application

Project Title

Catherine Do

Student Name

Ning Chen, Ph.D.

Advisor's Name

---

Advisor's signature

Date

---

Reviewer's Name

---

Reviewer's signature

Date

# **Abstract**

In this current age, computers are everywhere and it could be said that every student in the United States have access to one. Web-based applications provide the perfect solution to help students learn more about school subjects that they might be struggling on. Students can learn anytime about anything at anyplace as long as they have internet access. The EZ-Learning web application is designed to provide supplementary learning information on Biology for middle to college level students. Web-applications are great learning tools for students. It can provide extra help outside of the classroom for students that are struggling. There are websites for all sorts of subjects from coding to physics and math. However, there is a lack of free biology web-applications. Hence, my project with the EZ-Learning web application will provide the following:

1. A safe learning environment for students and teachers
2. Game prizes make it fun and rewarding for students to learn
3. Act as a bridge between students and teachers outside of the classroom

The development of this web application encompass the entire Software Development Life Cycle(SDLC) which includes gathering requirements, designing the right architecture for the system, implementation of the system using Agile methodologies, testing, and deployment. The completed product will provide a great learning tool for students and help teachers monitor their progress.

# Table of Contents

<b>Abstract</b>	<b>2</b>
<b>Keywords List</b>	<b>6</b>
<b>Introduction</b>	
<b>Rationale</b>	<b>7</b>
Background	8
Problem Statement	9
Project Goals and Objectives	9
Project Goals	9
EZ-Learning Objectives	9
Resources	9
<b>Software Requirements Specification (SRS)</b>	<b>10</b>
Introduction	10
Purpose	10
Project Scope	10
Overall Description	11
Product Perspective	11
User Classes Characteristics	11
Operating Environment	12
Design and Implementation constraints	12
Assumptions and Dependencies	12
Functional Requirements	12
Non-Functional Requirements (Quality Attributes)	17
Quality Attributes Scenarios	18
Use Case Diagram	20
Use Cases	21
<b>Design</b>	<b>31</b>
Architectural Diagram	31
Context Diagram	32
ASP.NET MVC Model	32
MVC Diagram	34
UML: Sequence Diagram	35
UML Class Diagram	36
Take Quiz Class Diagram	36
View Profile Class Diagram	37
<b>Implementation</b>	<b>37</b>

Product Backlog	38
Release Plan	39
Sprint 1	40
Backlog Refinement	40
Sprint Planning	41
Sprint Review Meeting	42
Sprint Burndown Chart	46
Sprint 2	47
Backlog Refinement	47
Sprint Planning	48
Sprint Review Meeting	49
Sprint Burndown Chart	58
Sprint 3	58
Backlog Refinement	59
Sprint Planning	60
Sprint Review Meeting	60
Sprint Burndown Chart	63
<b>Test and Integration</b>	<b>63</b>
Objectives	63
Approach	64
Test Tools	64
Test Environment	64
Hardware	64
Software	64
Test Cases	65
Test Results	68
TC-1: UI Testing of Registration process	68
TC-2: Performance Load Test	69
TC-3: Security test of valid and invalid inputs	70
TC-4: Error Handling	70
TC-5: Display Correct Quiz Score Calculations	71
<b>Conclusion</b>	<b>71</b>
<b>Bibliography</b>	<b>73</b>
<b>Appendix A: Installation Instructions</b>	<b>74</b>
Download Source Code:	74
Download Visual Studio	74
Installing SQL Server	74

Open WSAT	75
Run Application	75
<b>Appendix B: Operational Manual</b>	<b>75</b>
How to use the contact page	76
How to Register/Login	76
How to take the Quiz	77
How to view/update user's Profile	79

# Keywords List

Keywords	Definition	Related Terms
Application program interface	A tool to build software application	API
Cascading Style Sheets	A programming language for describing the presentation of a HTML document	CSS
Coded UI Tests	An automatic tool in Visual Studio to record UI testing	CUITs
Database management system	Software system for creating and managing databases	DBMS
Frame Rate per Second	Measurement of a device performance in displaying its content	FPS
Functional Requirement	Describes what the system should do	FR
GameMaker Language	Programming language used in GameMaker	GML
Graphical User Interface	A type of user interface that lets the user interacts with the computer	GUI
HyperText Markup Language	A programming language used to develop web pages	HTML
Internet Information Services	An extensible web server	IIS Express
Kanban	A method for managing software development	
Model view controller	A software architecture model for web applications	MVC
.NET framework .aspx	an open-source server-side web application framework that was developed by Microsoft for web development	ASP.NET
Non-functional Requirement	Describes how the system works	NFR
Software requirements	Contains the description of the software system	SRS

specification		
Scrum	An Agile framework for software development	
Scrumban	A combination of Scrum and Kanban frameworks	
Software development life cycle	Framework that defines tasks performed at each step in the software development	SDLC
Structured Query Language	Language for managing data in databases	SQL
Test Case	A set of conditions or variables that a tester will use to determine if the tested system satisfy the requirements and is working correctly	TC
Quality Attribute	factors that affect behavior, system design, and user experience	QA
User Interface	A way in which the user interacts with the system	UI
Use Case	Defines the interaction between a role and the system	UC
User Story	Description of a software feature	US
Web Site Administration Tool	A tool to manage users and roles	WSAT

## Introduction

### Rationale

The Interactive EZ-learning application web base application is created for teachers to give supplementary materials for students to learn. Since I was a Biology major, I want to combine my knowledge and love of Biology with Computer Science by developing a web based application that can teach other students. I want to use software engineering techniques that I have learned in class to develop the Interactive EZ-learning application for students from middle school to college students. This web application will let me practice my programming skills in conjunction with what I have learned as a Software Engineering student.

## Background

Interactive and e-learning is a digital approach to teaching students as compared to the traditional way which is learning in the classroom. It is becoming increasingly more popular in today's world due to the growth of the digital age [4]. Since the start of the year 2000, children are growing up with the knowledge of computers and learning with computers. This new generation of students consider computers as an extension of their brains [3]. This results in a shift in education and how things are taught.

In interactive learning, teachers play the role of a facilitator of knowledge rather than transmitting knowledge directly to students in a classroom [1]. Students can access information anywhere and participate with other students all around the world in online discussions. In addition, online interactive learning makes it easier to teachers to provide extra information and assess their student's progress in certain topics [2]. Interactive learning is useful in that it stimulates students mind outside of the classroom. Simulations are created through visual components that make topics more eye catching and encourages students to explore each modules through hyperlinks [10]. Furthermore, it is accessible anywhere, students can learn materials in the comfort of their homes. It provides a safe environment for students to learn. For instance, students that have trouble speaking outloud in class can use the interactive e-learning system to contact teachers or other students through emails.

In addition, interactive learning also increases student's ability to be more creative and innovative [3]. It teaches students to be more creative and innovative by inviting students to participate more actively through discussions and online materials that requires immediate answers rather than passively listening in a classroom. As a result, this may help students retain more knowledge.

## Problem Statement

Currently there are many interactive web based apps available. These interactive learning apps targets topics such as coding, math, reading, writing, and many more. Therefore, I want to develop the Interactive EZ-learning web app that will help students learn Biology. The Interactive EZ-learning app will be available for free, easy to use, and designed for middle school, high school, or college students. Its main focus will be as an interactive information provider for supplementary learning materials in Biology. It is a bridge between teachers and students outside of the classroom. The EZ-learning app provides an easy to use system that will help students learn more and allow teachers to monitor their progress.

## Project Goals and Objectives

### Project Goals

My goals for this project is to apply the software engineering techniques learned in this class to develop the EZ-Learning web application. This includes planning and designing the right architecture for this project and using Agile methodologies as my development model.

Apply software engineering techniques learned in class to develop my web base application. In addition, I want to improve my programming skills by implementing the web application using HTML 5, CSS3, jQuery, C# and JavaScript.

### EZ-Learning Objectives

My objectives for the EZ-learning system is to create a safe and dynamic website that is easy to use. It is a free learning source for students interested in learning more about Biology. I hope that the EZ-Learning system will become a bridge to connect students and teachers outside of the classroom and make it easier for students to learn and teachers to know which topic their students might be struggling on.

## Resources

### *Hardware*

Asus Laptop (Intel® Core™ i7 8.00GB RAM), running on Windows 10 on a 64bit operating system.

### ***Software and Developmental Tools***

I will be using Visual Studio 2015. I chose Visual Studio 2015 because it is free and it supports JavaScript, HTML5, C#, and ASP.NET.

### ***Programming Languages and Frameworks***

This application will be developed using HTML 5, CSS3, and JavaScript for the front-end web development process. C# shall be used for the back-end development process and mySQL will be the RDBMS for this application. ASP.NET as the framework for this application.

### ***Documentation***

The documentation of the application which is written in the final report is written on Microsoft Office 2016 (Word, Excel, PowerPoint). In addition, the diagrams are created on Visio 2013.

### ***Technologies:***

- ASP.NET framework
- SQL server
- jQuery library

## **Software Requirements Specification (SRS)**

### **Introduction**

#### **Purpose**

The purpose of this Software Requirements Specification (SRS) document is to provide a complete description of the requirements for the EZ-Learning web application. It will provide the project scope, the constraints, the functional and nonfunctional requirements, and the use cases of the system. The intended audience of this document is for all the stakeholders for this web application such as the software developers, the project managers, and the users.

#### **Project Scope**

The EZ-Learning learning web application is meant to supply supplementary biology information and tutorials for students. It will allow students from middle school to college level to access extra materials that will help them succeed in class. In addition, teachers will be able to log-in and monitor their student's progress by checking their quiz scores. The system will offer a fun and safe environment for students to learn by offering

students the options of downloading games when they have successfully passed a quiz. Furthermore, the system will be secure in making registration mandatory. Authentication is verified when users provide their password at login.

## Overall Description

### Product Perspective

This software product is intended for students to learning more about biology and for teachers to monitor their progress. Users must have an account to access this website. When a user registered for an account, the information entered is saved in a SQL server. Student users have a choice of 3 biology topics to cover and each topics have a quiz. At the completion of each quizzes, students are given a downloadable link for a game as a prize. The scores for each quizzes are saved in the database management system and can be viewed by the students or teachers.

### User Classes Characteristics

This web application is meant for students so the primary user is expected to be students from middle school to college level. Its secondary users are teachers. Student users are considered to be the main users, accounting for the majority of the users of this application. Since this is a free web application, the main purpose is to provide an educational website that will help students learn more about biology and succeed in class. Student users are expected to register for an account and login every time they access this website. Student quizzes are saved within the database in an SQL server so that their scores can be seen by teachers. In addition to quiz taking, students may download games when they pass a quiz. A passing grade is 4/5 or higher. Since each quizzes consists of 5 questions, a passing grade is given when the students successfully answers at least 4 out of 5 questions correctly.

Teachers must also have a registered account to use the EZ-Learning website. Teachers may view student scores and email students. Furthermore, teachers can view a student's progress by seeing which topic a student has successfully completed in viewing their quiz scores. This will help teachers understand which topics their students are struggling with and needed more help on.

## Operating Environment

EZ-Learning is a learning web application meant for students and teachers.

EZ-Learning operating environment will be on a web server so users must have internet access to use this web application. This application will be compatible across many different web browsers including: Google Chrome, Mozilla Firefox, Internet Explorer, Safari, and Opera. In addition, EZ-Learning will be available across different operating systems such as: Windows OS, Mac OS, Linux OS or any other operating system. Hardware components that are required for the implementation of EZ-Learning are computers, smart phones or tablets.

## Design and Implementation constraints

CO-1: All HTML codes shall adhere to the HTML 5.0 standard.

CO-2: The Navigational menu tool bar shall be available on all EZ-Learning web page.

CO-3: Access to view quiz scores are limited to only authorized users to protect students' privacy.

CO-4: The EZ-Learning app will store its data on a SQL server

## Assumptions and Dependencies

AS-1: Web server must be working 99.9999% of the time.

AS-2: Users have internet connection to access the application.

AS-3: Users have a valid email address for registration.

## Functional Requirements

Functional requirements are the requirements that are needed to make this application work Properly since it describe what the system is suppose to do.

### **FR-1: Log-in**

#### **Description**

User shall log-in for authorization of certain actions that requires a registered account

#### **Input**

- User press on log-in button
- User enter their username
- User enter their password
- User press log-in button

#### **Processing**

- System detects log in button action
- System queries database for user information

### **Output**

-User is redirect to homepage with account logged in

### **Error Handling**

-Username does not match password, system gives an error message and prompts the user to re-enter username and password again

-Username not detected, system shall prompt the user to register for an account

## **FR-2: Register Account**

### **Description**

Users shall be able to register for a student or a teacher account.

### **Input**

-User enter in their email address, password, re-entered password, and choose if they are a student or a teacher.

### **Processing**

-System shall check user's email address with database to ensure that no duplicate account was being created.

-System shall also detect that all required fields were filled out

-System shall check that the password and the re-entered password matched

-System shall store user's information in the database

### **Output**

-User is registered and shall be able to login their account

### **Error Handling**

-System shall prompt user to enter another email if a duplicate was used

-System shall display an error message if the passwords does not match and ask the user to re-enter it

## **FR-3: Attempt Quiz**

### **Description**

Only student users shall be able to attempt quizzes.

### **Input**

-User is signed in on a student account

-User clicks on "Take a Quiz" button

-User chooses a Biology topic to attempt Quiz

### **Processing**

-System detects button click

-System checks if user is logged in on a student account

-System prompts user for confirmation

-System redirect user to quiz page

### **Output**

- Quiz page is displayed

### **Error Handling**

- User not logged in, system shall prompt user to register or login

- System shall prompt user to finish the quiz if not all questions have an answer

## **FR-4: View Profile**

### **Description**

Students and teacher users are able to view their account profile with quiz scores

### **Input**

- User is logged in

- User have a profile created

### **Processing**

- System connects to the database

- System searches database

- System display the user's profile

### **Output**

- User's information is displayed

### **Error Handling**

- If there is no user information, the system shall prompt the user to set one up

## **FR-5: Leave message**

### **Description**

Students and teacher users can leave a message on the contact page which shall be stored in the database

### **Input**

- User is logged in

- User clicks on contacts at the navigation bar

- User fills out the form and click "finished" when completed

### **Processing**

- System display contact us page with form

- System shall check if user is logged in with finish button is clicked

- System stores message in database

### **Output**

- Message summary is displayed

- Message stored in database

### **Error Handling**

- User not logged in, system prompts user to login when user clicks on finish button

- Information is not completed, system prompts user to fill out all

## **FR-6: View Student progress**

### **Description**

Teacher user shall be able to view their student's progress and see which subject was completed

### **Input**

- Teacher user is logged into the system
- Teacher user clicks on progress report button

### **Processing**

- System checks if teacher user is logged in
- System queries database
- System displays students information

### **Output**

- Student's quiz scores, name, and date of completion is displayed

### **Error Handling**

-System shall prompt user to create an account or log into their account if the user is not signed in

## **FR-7: Download Game**

### **Description**

Student users shall be able to download games after successfully completing a quiz

### **Input**

- Student user attempts a quiz
- Student user completes quiz with a score of 70% or more

### **Processing**

- System checks score at completion of quiz
- System display link to download game if score is above a 70%

### **Output**

- Link to download game is displayed

### **Error Handling**

-If student user scores less than 70% on a quiz, the system shall prompt the user to try again and no download link is given

## **FR-8: View Biology Materials**

### **Description**

All users shall be able to view course materials without the need to sign in

### **Input**

- User clicks on a Biology topic button

### **Processing**

- System detects button click

-System direct user to the chosen Biology topic page

### **Output**

-Biology information is displayed

### **Error Handling**

-Material not shown, user is redirect to an error page with the message asking user to refresh the page, check if javascript is enabled, or clear cookies and cache

-Material shown appear differently than they should, user is redirect to error page asking user to check their browser extension or enable javascript

## **FR-9: View Game Information**

### **Description**

Any users shall be able to view the game information page to see the type of games that are available to be downloaded

### **Input**

-User clicks on “game” tab

### **Processing**

-System detects “game” button clicked

-System redirect user to game page

### **Output**

-Game page is displayed on the user’s browser

### **Error Handling**

-Material not shown, user is redirect to an error page with the message asking user to refresh the page, check if javascript is enabled, or clear cookies and cache

-Material shown appear differently than they should, user is redirect to error page asking user to check their browser extension or enable javascript

## **FR-10: View About Page**

### **Description**

Information about the EZ-Learning web application shall be shown on this web page to any users.

### **Input**

-User clicks on “about” tab

### **Processing**

-System redirect the user to the about page

### **Output**

-About page is displayed on the user’s browser

### **Error Handling**

-Material not shown, user is redirect to an error page with the message asking user to refresh the page, check if javascript is enabled, or clear cookies and cache

-Material shown appear differently than they should, user is redirect to error page asking user to check their browser extension or enable javascript

### **FR-11: View Quiz Scores**

#### **Description**

Students shall be able to view their quiz scores

#### **Input**

-User clicks on “about” tab

#### **Processing**

-System redirect the user to the about page

#### **Output**

-About page is displayed on the user’s browser

#### **Error Handling**

-Material not shown, user is redirect to an error page with the message asking user to refresh the page, check if javascript is enabled, or clear cookies and cache

-Material shown appear differently than they should, user is redirect to error page asking user to check their browser extension or enable javascript

## **Non-Functional Requirements (Quality Attributes)**

Non-functional requirements, also known as quality attributes describes how the system is built, designed, and implemented. The top four quality attributes that are most important to the EZ-Learning system is: Availability, Usability, Security, and Performance.

#### **NFR1: Availability**

Availability is one of the top quality attribute that the EZ-Learning system must have since users should be able to use the system anytime and anywhere. One of the main objective of this system is to provide supplementary biology lessons for students. Therefore, the system should be available 99.99% of the time.

#### **NFR2: Usability**

Usability is a very important quality attribute for the EZ-Learning system because the user should have an overall good experience with this application so that students would want to learn more and have an easier time in understanding biology.

#### **NFR3: Security**

Security is considered important because the system contains sensitive information such as the student’s quiz scores. Therefore, it is imperative that only users with the

correct access can see this information. Hence, the user shall only have 3 tries to enter the password correctly before the system places a hold on the account.

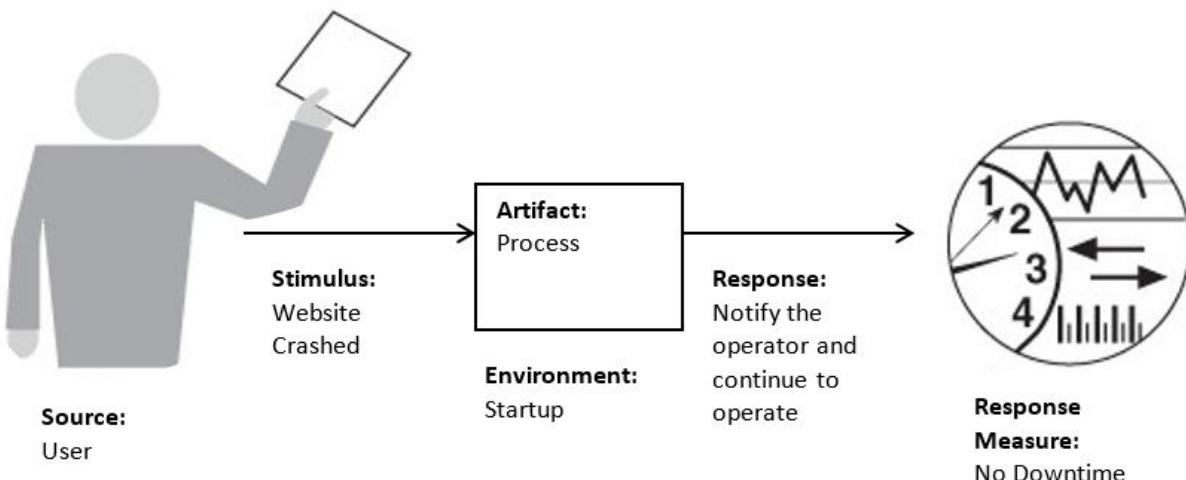
#### NFR4: Performance

Since the EZ-Learning system shall provide quizzes to test student's knowledge, it is important that the system's performance is responsive to the users requests.

#### Quality Attributes Scenarios

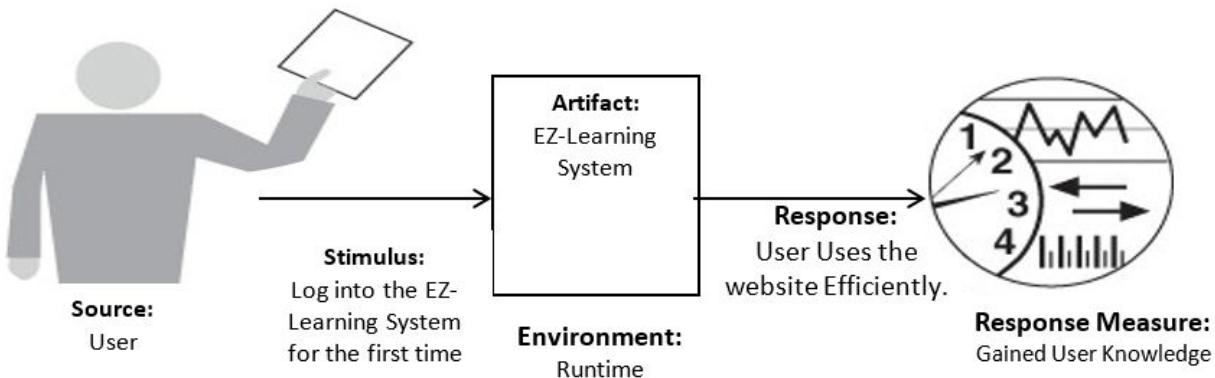
##### **QA 1: Availability**

1. A user reported that the EZ-Learning website crashed during startup. The system notified the operator and continues to operate with no downtime.



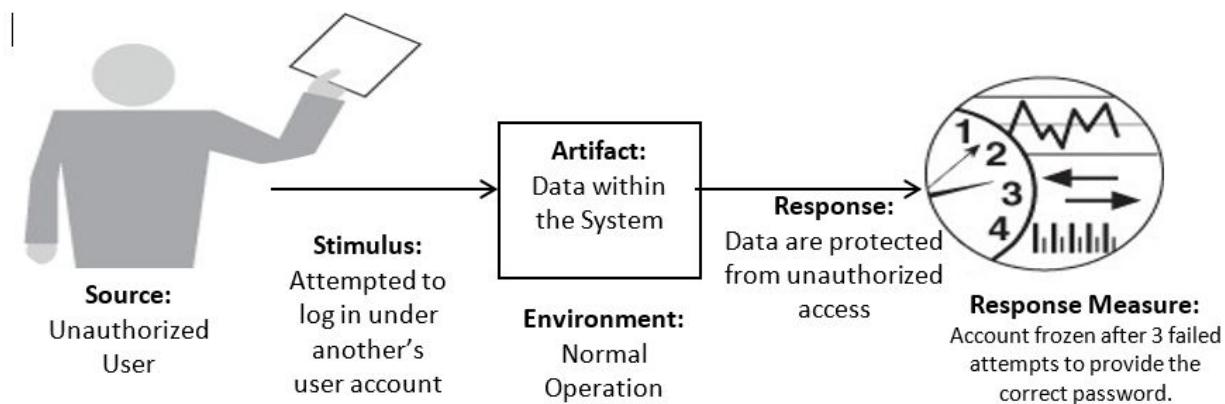
##### **QA 2: Usability**

1. The user logs into the EZ-Learning system for the first time and is exploring different layer of the navigation bar within the website. After using it efficiently for 30 minutes, the user gained new knowledge about the menu.



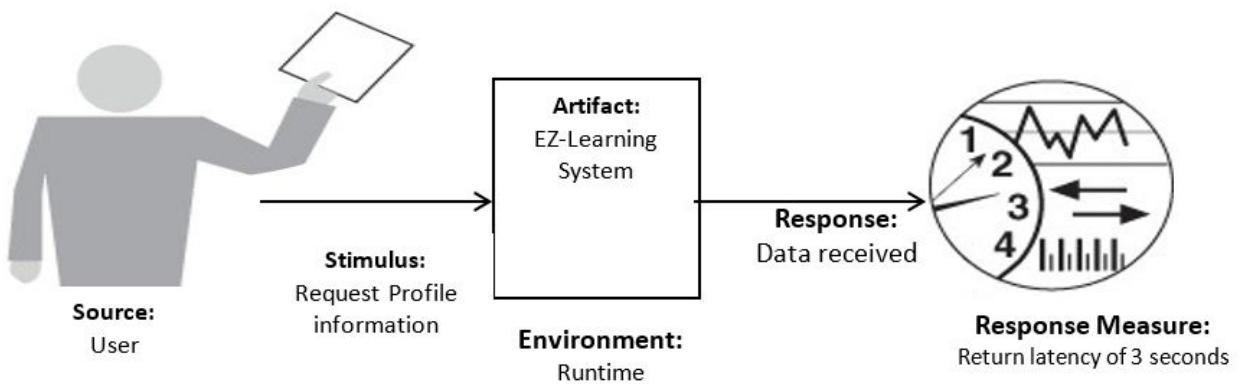
### QA 3: Security

1. An unauthorized user attempted to log in under another user's online banking account during normal operations. The system is protected from unauthorized access and froze the account after 3 failed attempts to provide the correct password and security questions at the login screen.



### QA 4: Performance

1. User requests profile information during normal operation. The system process the request with a latency of 3 seconds.



## Use Case Diagram

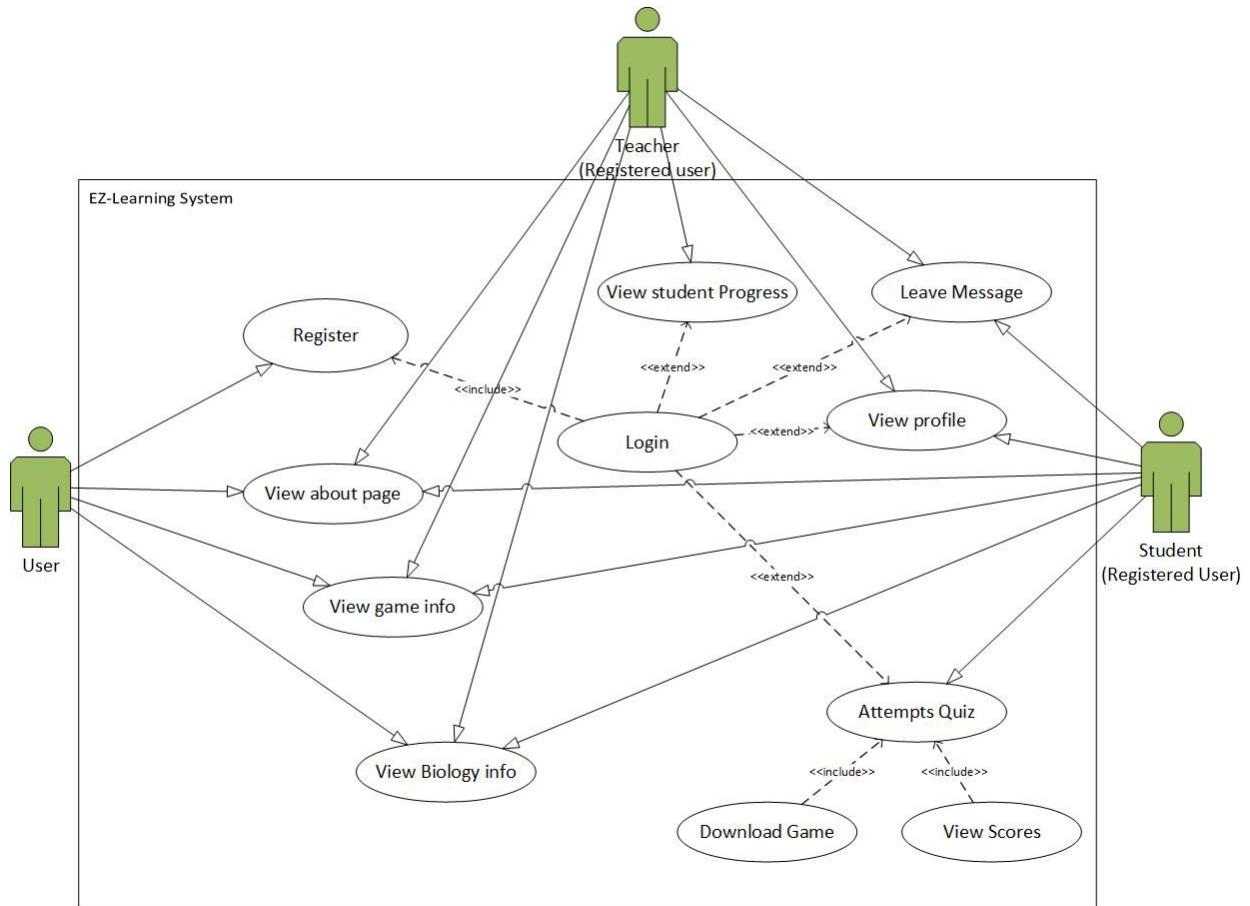


Figure 1: Use case Diagram

## **Use Cases**

### **Use Case Descriptions**

#### **Unregistered Users:**

Unregistered users are users without an account at the EZ-Learning web application. Users that do not have an account have limited access to the system.

#### **Registered Users:**

Registered users are users with an account at the EZ-Learning web application which grants special access depending on the type of registered users. The two main registered users consists of Student and Teacher users.

#### **Student Users**

Student users are users with a registered account at the EZ-Learning web application. Student users have all access of an unregistered users, registered users, and special access only available to student users.

#### **Teacher Users**

Teacher users are users with a registered account at the EZ-Learning web application. Teacher users have all access of an unregistered user, registered user, and special access only available to teacher users.

<b>Primary Actors</b>	<b>Use Cases</b>
Unregistered users	UC-2 Register for an account UC-8 View Biology materials UC-9 View game information UC-10 View about page
Registered user	UC-1 Log-in to account UC-5 Leave message UC-4 View profile
Student users	UC-3 Attempts Quiz UC-7 Download game UC-11 View Scores
Teacher users	UC-6 View Student Progress

<b>Use Case ID:</b>	<b>UC-1 :Log-In to Account</b>	
<b>Created By:</b>	Catherine Do	<b>Date Created:</b> March 4, 2017
<b>Actors:</b>	<b>Primary:</b> Registered User Database	<b>Secondary:</b>
<b>Description:</b>	User shall log-in by clicking on the login tab on the navigation bar.	
<b>Trigger</b>	User clicks on login button	
<b>Precondition</b>	1.User must have an account on the EZ-Learning application 2.User must have password	
<b>Postcondition</b>	User is logged in as either a student or teacher	
<b>Normal Flow</b>	1.0 Log-in to account 1.User clicks on login button on website 2.User input email and password 3.User clicks login button	
<b>Alternative Flow</b>	N/A	
<b>Exceptions</b>	1.0 E1 Invalid Email address 1.0 E2 Invalid password	
<b>Priority</b>	High	
<b>Frequency of Use</b>	About 100 uses per day	
<b>Assumptions</b>	User have a registered account and password at EZ-Learning.	

<b>Use Case ID:</b>	<b>UC-2 Register for an account</b>	
<b>Created By:</b>	Catherine Do	<b>Date Created:</b> March 4, 2017
<b>Actors:</b>	<b>Primary:</b> unregistered user <b>Secondary:</b> EZ-Learning System, database	
<b>Description:</b>	Users shall register for an account by click on the “Register” button. The user shall provide the following information: email address, password, and reconfirm the password by reentering it.	

<b>Trigger</b>	User clicks on register button
<b>Precondition</b>	1. User must have a valid email address 2. No duplicates of email address is in the system 3. Password must be at least 8 characters long
<b>Postcondition</b>	The account is created and the user's account information is stored in the system's database
<b>Normal Flow</b>	<p><b>2.0 Register for an account</b></p> <ol style="list-style-type: none"> <li>1. User click on Register button on EZ-Learning website</li> <li>2. User enter a valid Email address</li> <li>3. User enter a password that is at least 8 characters long</li> <li>5. User reenter the password and press Confirm button to complete registration</li> <li>6. System shall display a confirmation screen that tell the user that registration was completed successfully</li> </ol>
<b>Alternative Flow</b>	N/A
<b>Exceptions</b>	2.0 E1 Invalid Email address 2.0 E2 Password is not long enough 2.0 E3 Password does not match
<b>Priority</b>	High
<b>Frequency of Use</b>	Less than one hundred a day
<b>Assumptions</b>	User has a valid email

<b>Use Case ID:</b>	<b>UC-3 Attempt Quiz</b>	
<b>Created By:</b>	Catherine Do	<b>Date Created:</b> March 4, 2017
<b>Actors:</b>	<b>Primary:</b> Students	<b>Secondary:</b> EZ-Learning System
<b>Description:</b>	Users logged in as students shall attempt quizzes by clicking on the "Take a Quiz" button. There are three biology quiz topics that users can choose. Once a quiz option is chosen, the system shall ask the user for confirmation if the user wishes to proceed.	
<b>Trigger</b>	Student user clicks on "Take a Quiz" button on the EZ-Learning website.	

<b>Precondition</b>	1.User has a registered account 2.User is logged in as a student
<b>Postcondition</b>	1.EZ-Learning system shall display quiz scores at completion 2.Quiz score is stored in system's database 3.Link to download game is displayed if quiz score > 70%
<b>Normal Flow</b>	<b>3.0 Attempts Quiz</b> 1.User logs into student account 2.Student user clicks on take a quiz button 3.User selects a quiz topic 4.User confirms choice 5.User attempts quiz and clicks on finish button at completion 6.User confirms choice to finish quiz 7.System shall show student's quiz scores with a download link for a game if quiz score is over 70%
<b>Alternative Flow</b>	N/A
<b>Exceptions</b>	3.0 E1 Incomplete quiz 1.System prompts user to fill in all required field when finished button is clicked
<b>Priority</b>	High
<b>Frequency of Use</b>	Average of 50 uses per day
<b>Assumptions</b>	Users have a registered account and the correct plugins to view quizzes

<b>Use Case ID:</b>	<b>UC-4 View Profile</b>	
<b>Created By:</b>	<b>Catherine Do</b>	<b>Date Created:</b> March 4, 2017
<b>Actors:</b>	<b>Primary:</b> Registered Users	<b>Secondary:</b> EZ-Learning System
<b>Description:</b>	Registered users shall be able to view their profile by clicking on the "My Account" tab once they are signed into their account.	
<b>Trigger</b>	User clicks on My Account tab	
<b>Precondition</b>	1.User must have a registered account 2.User must be logged into their account	

<b>Postcondition</b>	The user can view their profile information inside the My account page
<b>Normal Flow</b>	<p><b>4.0 View Profile as a Student user</b></p> <ol style="list-style-type: none"> <li>1. User logs into account</li> <li>2. User clicks on My account tab</li> <li>3. System displays user's profile</li> <li>4. User profile shows option to view quiz scores</li> </ol>
<b>Alternative Flow</b>	<p><b>4.1 View Profile as a Teacher user</b></p> <ol style="list-style-type: none"> <li>1. User logs into account</li> <li>2. User clicks on My account tab</li> <li>3. System displays user's profile</li> <li>4. User profile shows option to view student's progress</li> </ol>
<b>Exceptions</b>	<p>4.0 E1 Account not created</p> <ol style="list-style-type: none"> <li>1. System prompts user to sign in</li> </ol>
<b>Priority</b>	Normal
<b>Frequency of Use</b>	Less than 100 uses per day
<b>Assumptions</b>	Users have access to the EZ-Learning website with a registered account

<b>Use Case ID:</b>	<b>UC-5 Leave Message</b>	
<b>Created By:</b>	<b>Catherine Do</b>	<b>Date Created:</b> March 4, 2017
<b>Actors:</b>	<b>Primary:</b> Registered User	<b>Secondary:</b> EZ-Learning System
<b>Description:</b>	Messages can be left by registered users at the contacts page. Users must fill out the entire form to leave a message at the contact page.	
<b>Trigger</b>	User clicks on "contacts" tab	
<b>Precondition</b>	1. User must have a registered account 2. User must be logged into their account	
<b>Postcondition</b>	A message confirmation is displayed at completion and the message is saved in the system's database	
<b>Normal Flow</b>	<p><b>5.0 Leave Message</b></p> <ol style="list-style-type: none"> <li>1. Logged in user clicks on Contacts tab</li> </ol>	

	<p>2.User enters their first and last name with a valid phone number and contact preference on the first page</p> <p>3.User clicks next then complete the second page by checking the product that they are interested in</p> <p>4.User shall be able to leave a message on the comment box and then clicks finish</p> <p>5.System displays a summary of the inputted information</p> <p>6.System shall store this information in the database</p>
<b>Alternative Flow</b>	N/A
<b>Exceptions</b>	<p>5.0 E1 Invalid Email address</p> <p>5.0 E2 Invalid first and last name</p> <p>    1.system prompts user to enter valid first and last name with characters only</p> <p>5.0 E3 Invalid phone number</p> <p>    1.phone number inputted must all be numerical</p>
<b>Priority</b>	Normal
<b>Frequency of Use</b>	Averages 20 uses per day
<b>Assumptions</b>	Users have a valid email and phone number

<b>Use Case ID:</b>	<b>UC-6 View Student Progress</b>	
<b>Created By:</b>	<b>Catherine Do</b>	<b>Date Created:</b> March 4, 2017
<b>Actors:</b>	<b>Primary:</b> Teacher	<b>Secondary:</b> Database
<b>Description:</b>	Teacher users shall be able to view student's progress from a link inside their profile page	
<b>Trigger</b>	User clicks on Student Progress link inside their profile page	
<b>Precondition</b>	<p>1.User is logged in as a teacher</p> <p>2.Student had completed a quiz</p> <p>3.Quiz scores are stored in the database</p>	
<b>Postcondition</b>	Student's email and quiz scores are displayed	
<b>Normal Flow</b>	<p><b>6.0 View Student Progress</b></p> <p>1.User is logged in as a teacher</p> <p>2.User clicks on My account tab</p> <p>3.User clicks on student progress link inside their profile page</p>	

	4.System displays student's email and quiz scores
<b>Alternative Flow</b>	N/A
<b>Exceptions</b>	6.0 E1 Quiz scores not available 1.System display error message that a student must complete a quiz first for quiz scores to be available
<b>Priority</b>	Normal
<b>Frequency of Use</b>	Less than 100 per day
<b>Assumptions</b>	Student(s) have completed a quiz and received a score

<b>Use Case ID:</b>	<b>UC-7 Download Game</b>	
<b>Created By:</b>	<b>Catherine Do</b>	<b>Date Created:</b> March 4, 2017
<b>Actors:</b>	<b>Primary:</b> Student	<b>Secondary:</b> EZ-Learning System
<b>Description:</b>	Students user shall be able to download a game when they have successfully completed a quiz with a score > 70%. A link to download the game shall be given at the end of the quiz.	
<b>Trigger</b>	User completed a quiz with a score > 70%	
<b>Precondition</b>	1.User is signed in as a student user 2.User completes a quiz	
<b>Postcondition</b>	Download link is given when the user scores above 70%	
<b>Normal Flow</b>	<b>7.0 Quiz completed with a score &gt; 70%</b> 1.user is signed in as a student 2.user chooses a quiz topic and attempts it 3.user completes quiz with at least 4 out of 5 questions correct 4.System displays quiz score and download link for a game 5.user clicks on link to begin download	
<b>Alternative Flow</b>	<b>7.1 Quiz completed with a score &lt; 70%</b> 1.user is signed in as a student 2.user chooses a quiz topic and attempts it 3.user completes quiz with more than one questions incorrect 4.system display quiz score and prompts user to try again	
<b>Exceptions</b>	7.0 and 7.1 E1 user is not signed in as a student	

	7.1 E1 Internet connection not fast enough to download
<b>Priority</b>	Normal
<b>Frequency of Use</b>	Averages about 50 uses per day
<b>Assumptions</b>	User is signed into a student account with access to the EZ-Learning website and internet connection fast enough to handle the download

<b>Use Case ID:</b>	<b>UC-8 View Biology Materials</b>
<b>Created By:</b>	<b>Catherine Do</b> <b>Date Created:</b> March 4, 2017
<b>Actors:</b>	<b>Primary:</b> Unregistered user <b>Secondary:</b> EZ-Learning System
<b>Description:</b>	Any users shall be able to view biology information on the EZ-Learning website by clicking on a biology topic(DNA information, Protein Synthesis, Cell Biology) that they are interested in.
<b>Trigger</b>	User clicks on a biology topic
<b>Precondition</b>	User have access to the EZ-Learning website
<b>Postcondition</b>	Biology information is displayed on the user's browser
<b>Normal Flow</b>	<p><b>8.0 View Biology Materials</b></p> <p>1.User clicks on a biology topic tab      2.The system redirects the user to the chosen biology topic page      3.Biology information is displayed on the user's browser</p>
<b>Alternative Flow</b>	N/A
<b>Exceptions</b>	8.0 E1 System cannot load the information
<b>Priority</b>	High
<b>Frequency of Use</b>	Average about 300 uses per day
<b>Assumptions</b>	User have internet access

<b>Use Case ID:</b>	<b>UC-9 View Game Information</b>
<b>Created By:</b>	<b>Catherine Do</b> <b>Date Created:</b> March 4, 2017
<b>Actors:</b>	<b>Primary:</b> Unregistered user <b>Secondary:</b> EZ-Learning System
<b>Description:</b>	Users shall be able to view the games that are available to download once a quiz is passed with a score of 70% or more. The game information page includes screenshots of the games and a short description about it.
<b>Trigger</b>	User clicks on Games tab
<b>Precondition</b>	User must have access to the EZ-Learning website
<b>Postcondition</b>	Game information is displayed on the user's browser
<b>Normal Flow</b>	<b>9.0 View Game information</b> 1.User clicks on game tab 2.System redirect user to game information page 3.Game information is displayed on the user's browser
<b>Alternative Flow</b>	N/A
<b>Exceptions</b>	8.0 E1 System cannot load the information
<b>Priority</b>	Normal
<b>Frequency of Use</b>	Average about 200 uses per day
<b>Assumptions</b>	User have internet access

<b>Use Case ID:</b>	<b>UC-10 View About Page</b>
<b>Created By:</b>	<b>Catherine Do</b> <b>Date Created:</b> March 4, 2017
<b>Actors:</b>	<b>Primary:</b> Unregistered user <b>Secondary:</b> EZ-Learning System
<b>Description:</b>	Users shall be able to view the about page to learn more about the EZ-learning website
<b>Trigger</b>	User clicks on About tab
<b>Precondition</b>	User must have access to the EZ-Learning website

<b>Postcondition</b>	Information about the EZ-Learning website is displayed
<b>Normal Flow</b>	<b>10.0 View About Page</b> 1.User clicks on About tab 2.System redirect user to About page 3.Information about the EZ-Learning website is displayed on the user's browser
<b>Alternative Flow</b>	N/A
<b>Exceptions</b>	8.0 E1 System cannot load the information
<b>Priority</b>	Normal
<b>Frequency of Use</b>	Average about 200 uses per day
<b>Assumptions</b>	User have internet access

<b>Use Case ID:</b>	<b>UC-11 View Quiz Scores</b>	
<b>Created By:</b>	<b>Catherine Do</b>	<b>Date Created:</b> March 4, 2017
<b>Actors:</b>	<b>Primary:</b> Student	<b>Secondary:</b> Database
<b>Description:</b>	Student users shall be able to view their quiz scores after a quiz is completed	
<b>Trigger</b>	Student clicks on Quiz link in their profile page	
<b>Precondition</b>	1.Must be registered as a student 2.At least one quiz must be completed and a score is given	
<b>Postcondition</b>	Quiz scores are displayed	
<b>Normal Flow</b>	11.0 View Quiz Scores 1.User logged in to student account 2.User clicks on My Account tab 3.User clicks on Quiz link in profile page 4.System display quiz scores on user's browser	
<b>Alternative Flow</b>	N/A	
<b>Exceptions</b>	8.0 E1 System cannot load the information 8.0 E2 No quiz score was recorded	

<b>Priority</b>	High
<b>Frequency of Use</b>	Averages about 100 uses per day
<b>Assumptions</b>	User has a registered student account and had completed a quiz with a given score

## Design

### Architectural Diagram

The architectural design for the EZ-Learning web application follows the three tier architecture pattern (Presentation, Logic, and Data). The first tier is the front-end web server which is the content rendered by the browser. The middle tier which consists of the ASP.NET application server. Finally, the third tier is the back-end tier which is the SQL Server Database that manages and stores the data.

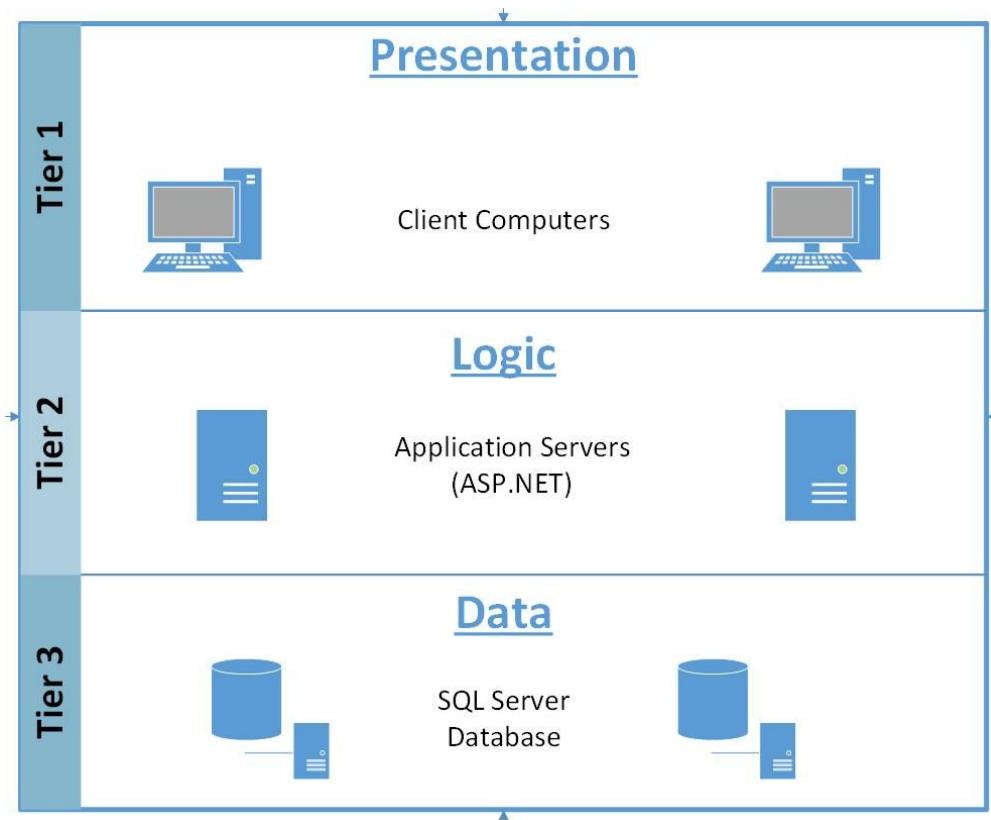


Figure 2: 3-Tier architecture design for the EZ-Learning web application

## Context Diagram

A system context diagram is a high level view of a system which shows the boundary between the system and its environment such as the entities that interact with it.

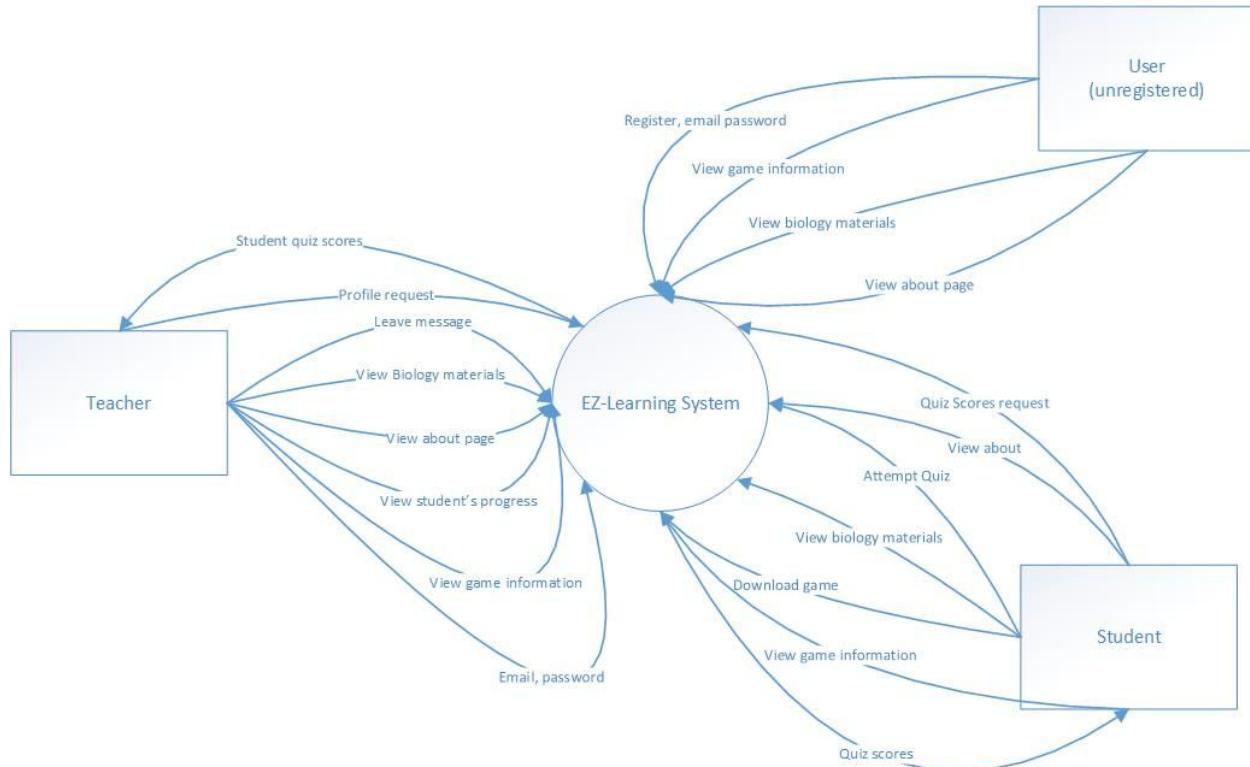


Figure 3: Context Diagram

## ASP.NET MVC Model

The EZ-Learning web application is created using ASP.NET MVC architectural model. The MVC model shows the architectural pattern of the User and UI interaction of the application. The MVC pattern is separated into three parts: the model, view, and controller. The model handles the data and fundamental behaviors of the application such as requests for information, response to instructions, or notifying observers in event-driven systems when information changes. The view render data from the model to form the user interface and it is what the user sees. The controller receives user inputs and calls the repository.

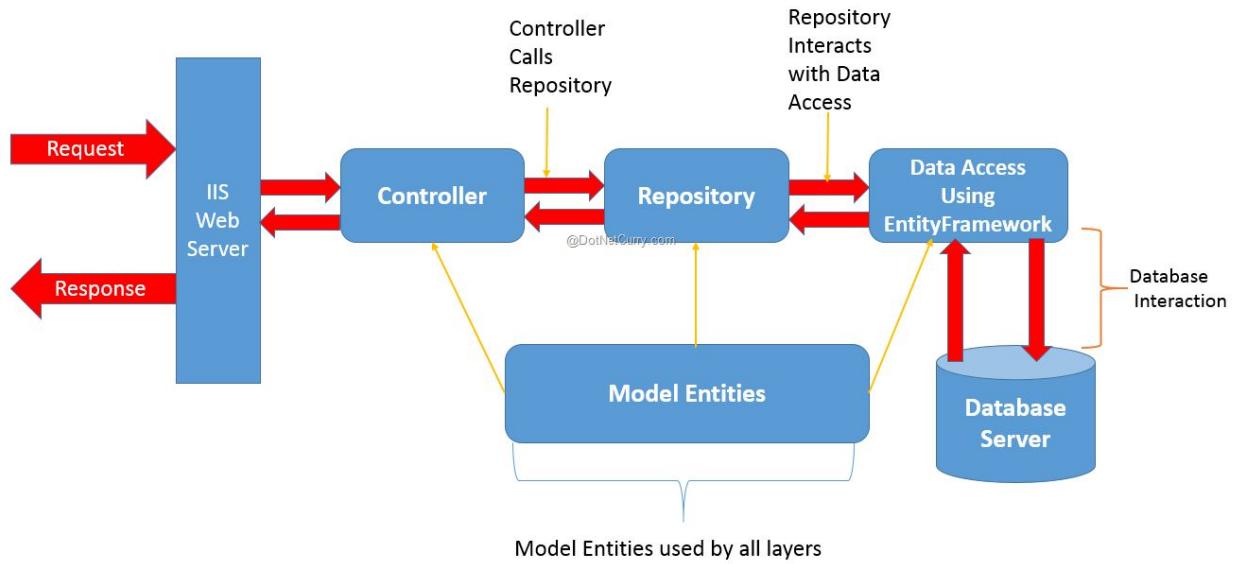


Figure 4: MVC diagram for ASP.NET. Source: Sabinis Mahesh. ASP.NET MVC 5: Using a Simple Repository Pattern for Performing Database Operations [14].

## MVC Diagram

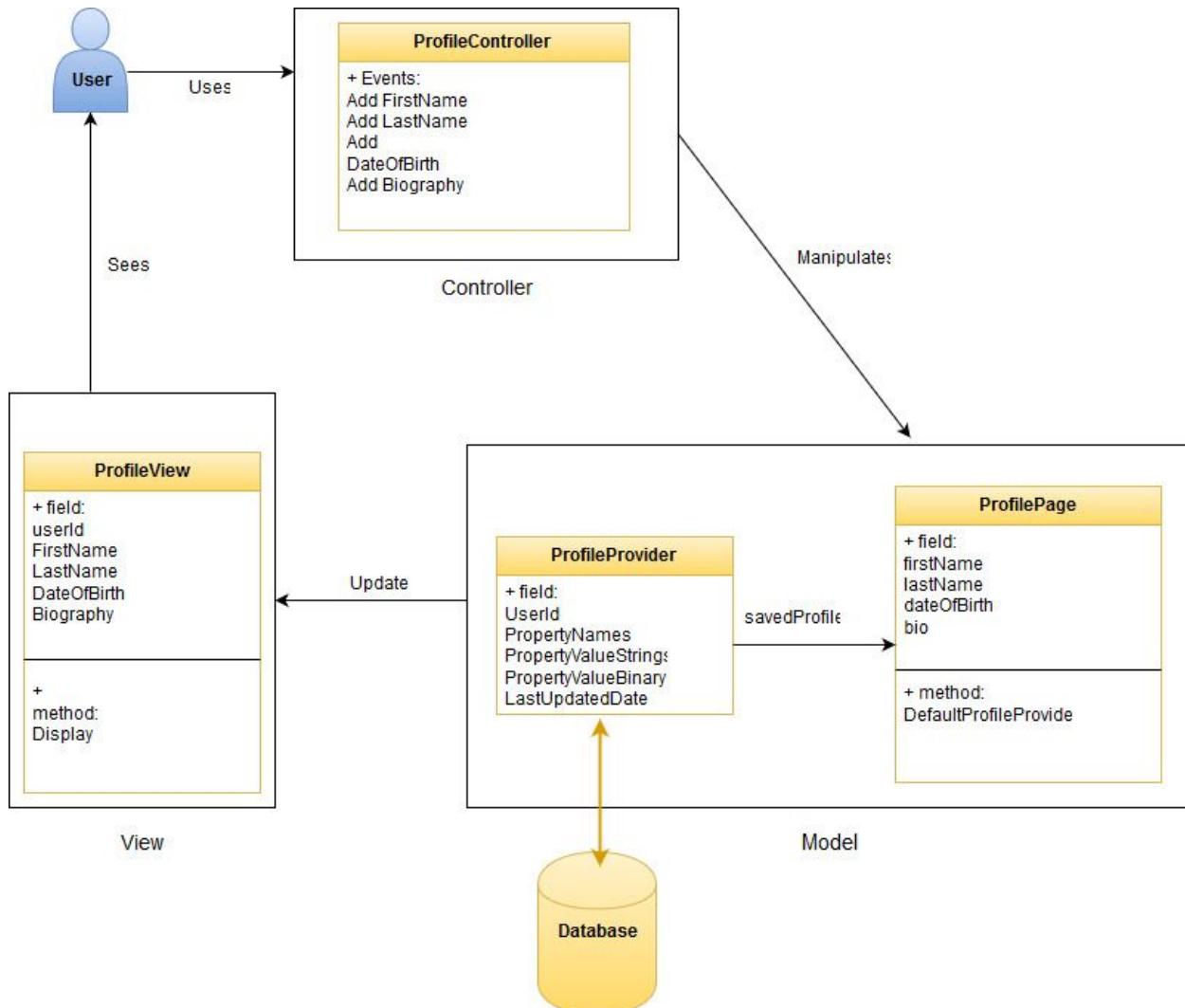


Figure 5: MVC Diagram

The Model View Controller diagram for updating user profile. The user sends a request to the web server which is then passed to the controller.

## UML: Sequence Diagram

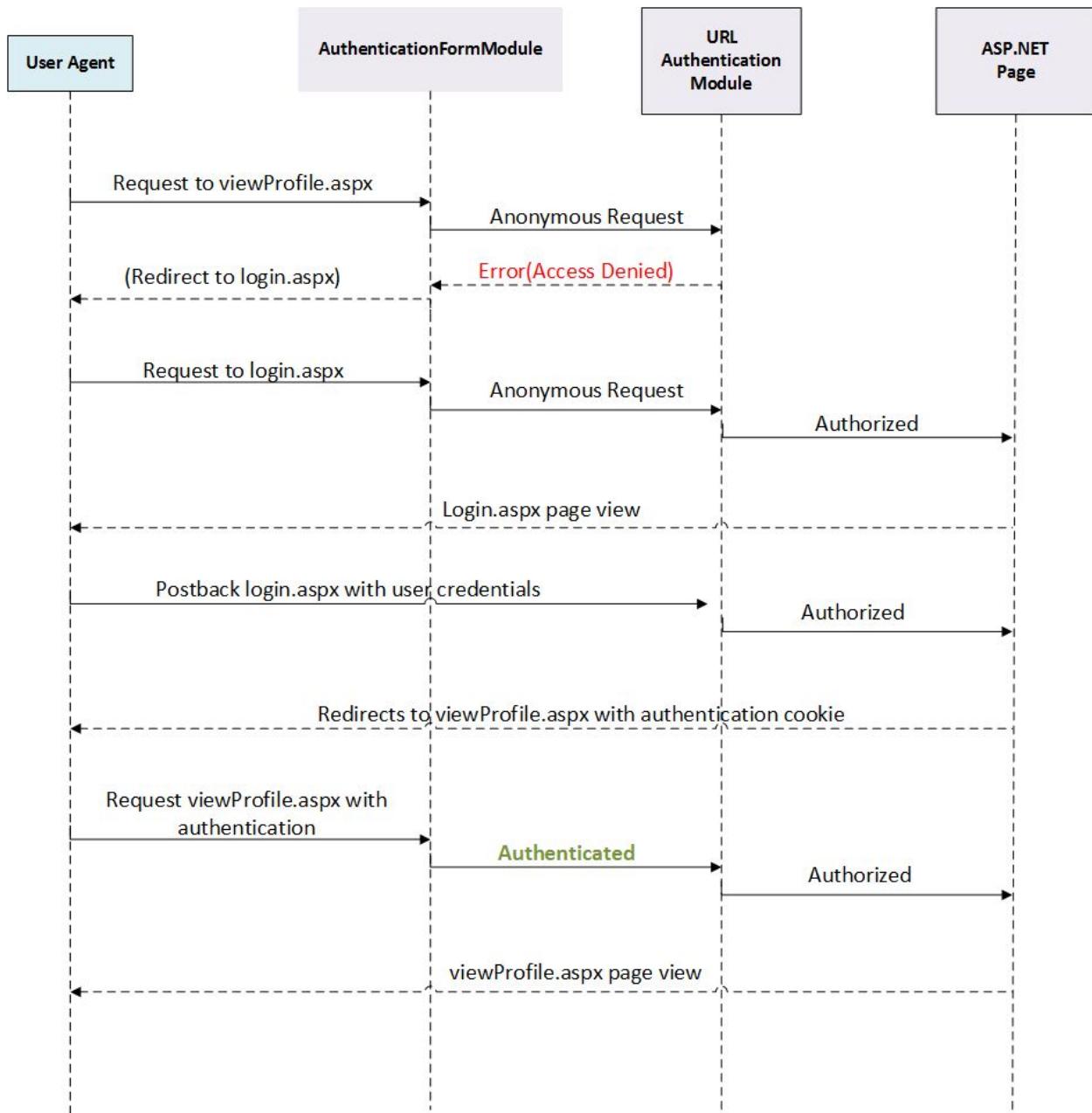


Figure 6: UML sequence diagram to view the profile page

The sequence diagram shows the interactions between the different processes. In this diagram, it shows a sequence of events for the user to view the profile page. The user can only view their profile page if the user is logged into their account since the profile page is protected from anonymous access.

# UML Class Diagram

## Take Quiz Class Diagram

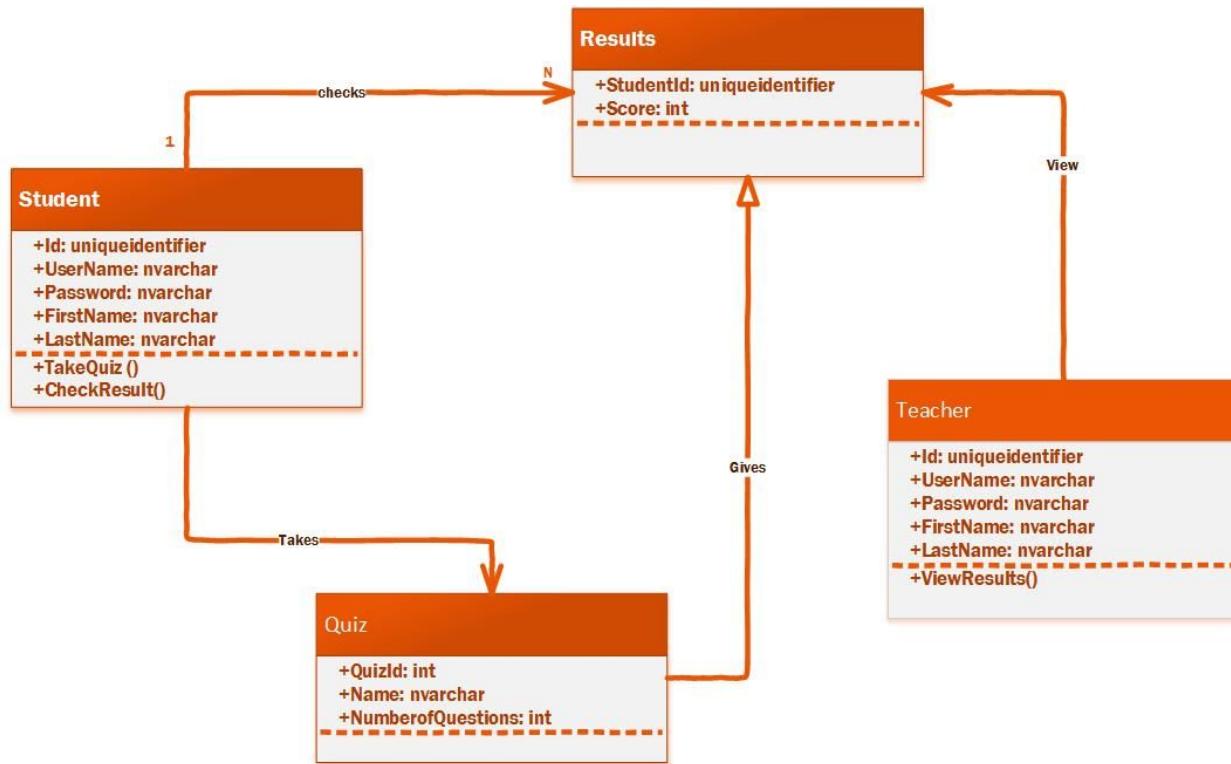


Figure 7: Quiz Class diagram

This class diagram shows the attributes and operations in each classes and their relationships. The Student class can check the results class containing the attributes: student ID and scores. The student class can also take a quiz which contains the attribute quiz id, quiz name, and the number of questions in the quiz. Meanwhile, the teacher class can view the results class which came from the quiz class.

## View Profile Class Diagram

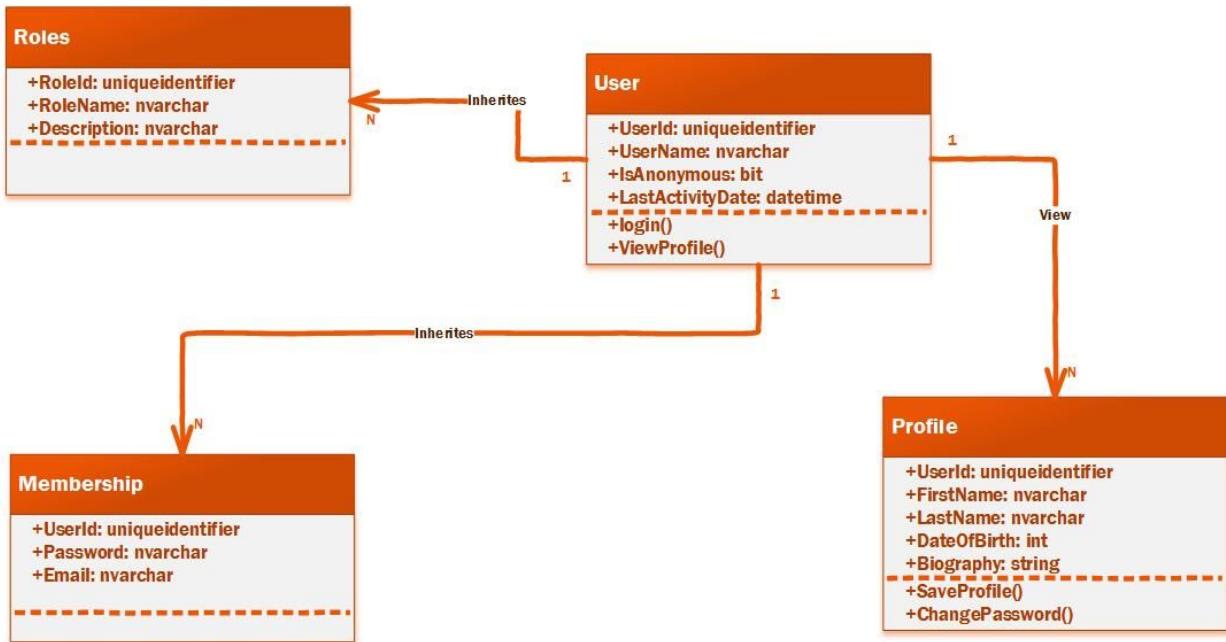


Figure 8: View Profile Diagram

The View Profile Class diagram shows the relationships between the roles, user, profile, and memberships classes. The User class inherits the roles and membership classes containing the attributes that makes up the user class. In addition, student class can also contain the profile class when the user sets up a profile.

## Implementation

The EZ-Learning web based application shall be developed using a Scrumban approach which is a combination of Scrum and Kanban. To limit the amount of excessive overhead work, only a part of Scrum ceremonies shall be documented. For instance, there is no daily scrum meetings. However, a sprint planning activity shall be conducted at the start of each sprint to produce a sprint backlog. Furthermore, at the end of the sprint, a sprint review is conducted to present what was accomplished during

that sprint. In addition, a Kanban board(Trello) shall be used to visualize the workflow. The artifacts generated for each sprint shall be as followed:

- Backlog Refinement
- Sprint Backlog
- Sprint Burndown Chart

## Product Backlog

The names and descriptions for each tasks as given by the product owner is feature in this prioritized product backlog. The tasks have been prioritized by the Product Owner. Each tasks are assigned a story points based upon the complexity of the task. These story points acts as a relative measurement for developers to understand the perceived difficulty of implementing each tasks.

### Major Features

FR-1: Log-in  
FR-2: Register Account  
FR-3: Attempt Quiz  
FR-4: View Profile  
FR-5: Leave message  
FR-6: View Student progress  
FR-7: Download Game  
FR-8: View Biology Materials  
FR-9: View Game Information  
FR-10: View About Page  
FR-11: View Quiz Scores

Name	Description	Story Points
1.View Homepage	As a user, I can view the homepage so that I can navigate to other parts of the website	4
2.View Biology Materials	As a user, I can view the biology materials so that I can understand more about Biology	4
3.View Game Information	As a user, I can view the game information so that I will understand what games are available	4
4.View About Page	As a user, I can view the about page so that I can know more about the EZ-Learning website	4
5.Download Games	As a student, I can download a game after	18

	completing a quiz with a 70% or higher so that I can get my reward for passing.	
6.Register for an Account	As a user, I can register for an account so that I can have more access to other portions of the website	10
7.Log-in to an Account	As a user, I can log in to my account so that I can perform certain functions that have restricted access.	10
8.Leave a Message	As a user, I can leave a message on the contact page so that I can provide feedbacks to make this website better.	8
9.View Profile	As a user, I can view my profile once I have registered for an account so that I can check for more information	4
10.Attempt Quiz	As a student, I can attempt to take a quiz so that I can prove that I understand the biology materials given to me on the website	14
11.View Quiz Scores	As a student, I can view my quiz scores so that I can know what topics that I need to improve on and which ones I did well on.	9
12.View Student Progress	As a teacher, I can view the progress of the students in my class so that I can see what scores they received on the quizzes.	9

## Release Plan

The product backlog have a rough estimate with the story points which was prioritized by the Product Owner. In order to finish the project in 3 sprints, I have set the max capacity for each sprint to be 45 story points. Therefore, each sprint should be an average of 33 story points to reach the goal.

	Sprint 1	Sprint 2	Sprint 3
	View About page (4)	Log-in to an account (10)	View quiz scores (9)
	View Game information (4)	Register for an account(10)	Attempt Quiz (14)

	Download Games (18)	Leave Message (contact page) (8)	View student progress(9)
	View Biology materials (4)	View profile (4)	
	View Homepage (4)		
<b>Plan</b>	34	32	32
<b>Capacity</b>	45	45	45

## Sprint 1

Timebox: 10days (February 2 - 12)

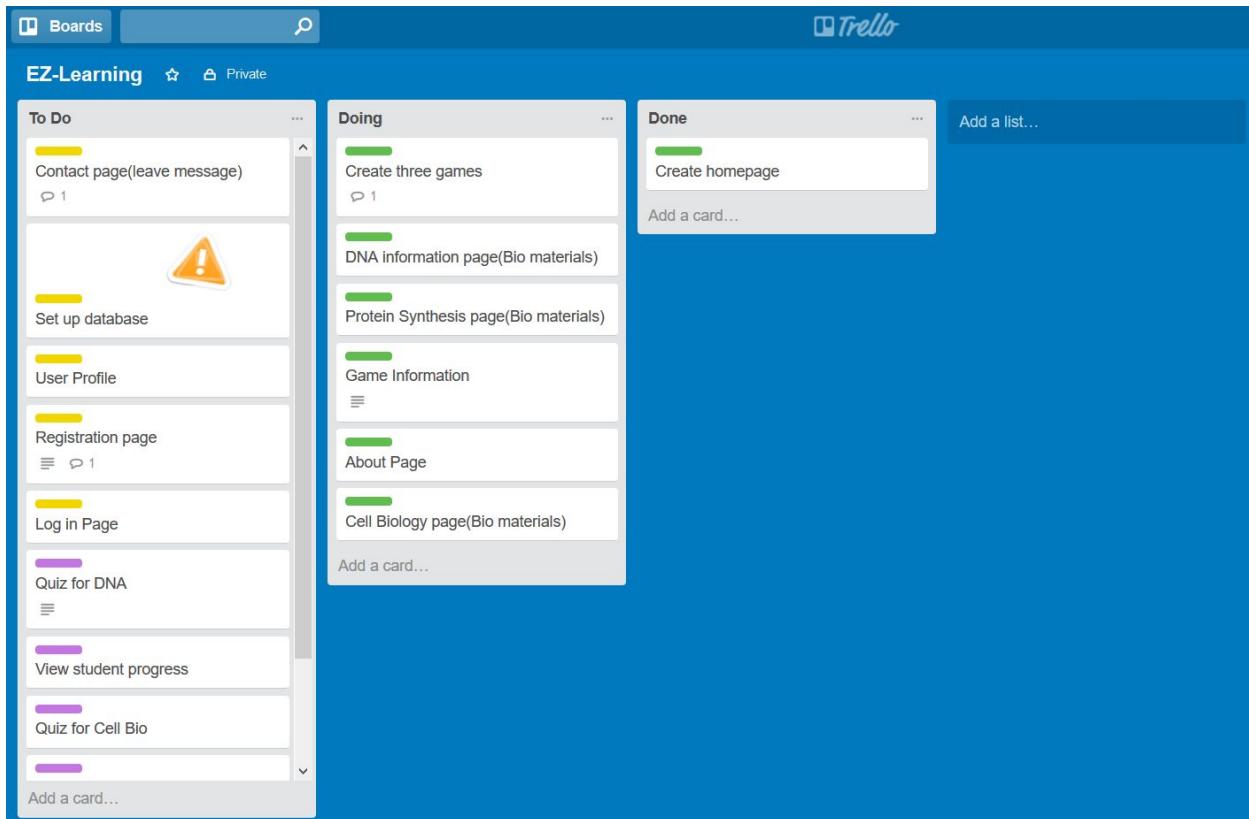


Fig 9 : Kanban board in Trello

## Backlog Refinement

The user stories(US) for this sprint is prioritized and the acceptance criteria for each US is defined. For the first sprint, US1 - US5 were implemented.

#### **US-1: View Homepage**

- Layout created with 100% functional navigation links
- Consistent-looking pages with master page
- Images and banner is displayed
- Bread crumb trail to enable user to easily navigate website

#### **US-2: View Biology Materials**

- Course materials are shown in correct formatting
- Page is consistent with other pages from website
- Images are displayed

#### **US-3: View Game Information**

- Three games information is displayed with correct formatting
- Page is consistent with other pages from website
- Images are displayed

#### **US-4: View About Page**

- Information is displayed with correct formatting
- Page is consistent with other pages from website

#### **US-5: Download Games**

- Games are 100% functional
- Game links allow user to download an exe file
- Game link only appears when user score a 4 more greater on Quizzes

### **Sprint Planning**

The goals for sprint 1 is to set up the basic frontend development side of the website so that the users can access certain information such as the homepage, biology materials, game information, and information about the website.

#### **Sprint Backlog**

##### **US-1: View Homepage**

###### **Tasks**

- Design/Implement Page layout
- Utilize Navigation Controls
- Inherits from masterpage

##### **US-2: View Biology Materials**

###### **Tasks**

- Design/Implement Page layout
- inherits from masterpage
- Add relevant biology information with pictures

##### **US-3: View Game Information**

### Tasks

- Design/Implement Page layout
- Inherits from masterpage
- Add game control information and screenshots

### US-4: View About Page

#### Tasks

- Design/Implement Page layout
- Inherits from masterpage

### US-5: Download Games

#### Tasks

- Development of a TicTacToe game using GameMaker
- Development of Galactic Mail game using GameMaker
- Development of Virus Attack game using GameMaker
- Integrate download link for games into webpage

## Sprint Review Meeting

The purpose of the sprint review meeting is to summarize what was accomplished in the sprint and to demonstrate the product. The main focus on sprint 1 was to implement a homepage with links to other informational pages on the website. In addition, work was started on the development of the three games that is supposed to act as rewards for students who had successfully completed a quiz.

The screenshot shows the homepage of the EZ-Learning website. At the top, there is a banner featuring a collage of biological images: green bacteria-like cells on the left, a blue molecular structure in the center, and a red neuron on the right. Overlaid on the center of the banner is the word "EZ-LEARNING" in large, bold, white capital letters. To the right of the banner is a navigation bar with links to "Home", "About", "Login", and "Contacts". Below the banner is a sidebar on the left containing links: "DNA Information", "Protein Synthesis", "Cell Biology", "Take a Quiz", and "Games". The main content area has a header "Introduction" and a paragraph of text explaining the website's purpose. To the right of the text is a small image of a DNA double helix. Below the text is a larger, more detailed image of a DNA double helix.

**Home**

**Introduction**

Biology is the study of life and living organisms, from the smallest bacteria to giant sequoias. Biologists use observation and experimentation to gain an understanding about the natural world. Branches of biology include anatomy, biotechnology, botany, cell biology, ecology, genetics, medicine, microbiology, molecular biology, and zoology. Many people entering the field of biology become specialized in a particular area.

The website is a resource site for science teachers and students who wants to learn more in biology. It contains a variety of lessons, quizzes/tests, interactive games, worksheets, and information on science topics for all levels. You can find lessons related to biology topics in the links listed on the sidebar.



Three important topics in biology are covered in the website. Students will learn interesting information about DNA such as how DNA replicates as see in the picture to the left. In addition, this website also contain information about protein synthesis. The last topic is about the basic animal cells and plant cells structures.

Fig 10: Homepage for the EZ-Learning website

The homepage of the EZ-Learning application in the first sprint with links to other pages of the website (fig.10). Only links to DNA information, Protein synthesis, Cell Biology, Games, Home, and About have actual contents in those pages as the other links are only placeholders(fig.11).

```

<title>My Website</title>
<%--<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/animate.css/3.5.2/animate.min.css"/> <!--To use an an
<asp:ContentPlaceHolder id="head" runat="server">
</asp:ContentPlaceHolder>
<script src="/Scripts/modernizr-2.7.1.js"></script>

</head>
<body>

<form id="form1" runat="server">
    <div id="nav">
        <asp:Menu ID="Menu1" runat="server" CssClass="nav" DataSourceID="SiteMapDataSource1" Orientation="Horizontal" St
            BorderStyle="None"></asp:Menu>
        <asp:SiteMapDataSource ID="SiteMapDataSource1" runat="server" ShowStartingNode="False" />
    </div>
    <header>
        
    </header>

    <div id="sidenav">
        <!--Side navigation bar--->
        <ul>
            <li><a href="DNA.aspx">DNA Information</a></li>
            <li><a href="protein.aspx">Protein Synthesis</a></li>
            <li><a href="cell.aspx">Cell Biology</a></li>
            <li><a href="quiz.aspx">Take a Quiz</a></li>
            <li><a href="games.aspx">Games</a></li>
        </ul>
    </div>

```

Fig 11: Masterpage in ASP.NET coded with HTML5

All web pages in this application inherits from the masterpage for a consistent look(fig. 12).

```

<%@ Page Title="DNA" Language="C#" MasterPageFile="~/MasterPages/Frontend.master" AutoEventWireup="true" CodeFile="DNA.aspx.cs"
    Inherits="DNA" %>

```

Fig 12. An example of the codes from DNA page which shows inheritance from Masterpage

The last user story for sprint 1 was to complete three games for the users to download when the passed a quiz with a score greater than 4. Although the goal was three completed games, only two were completed at the end of sprint 1. The two games were TicTacToe and Galactic Mail (fig 13).



Fig 13: Games created in GameMaker

Galactic Mail developed in GameMaker using free sprites provided by GameMaker. This game was created using GameMaker drag and drop features to code (fig. 14).

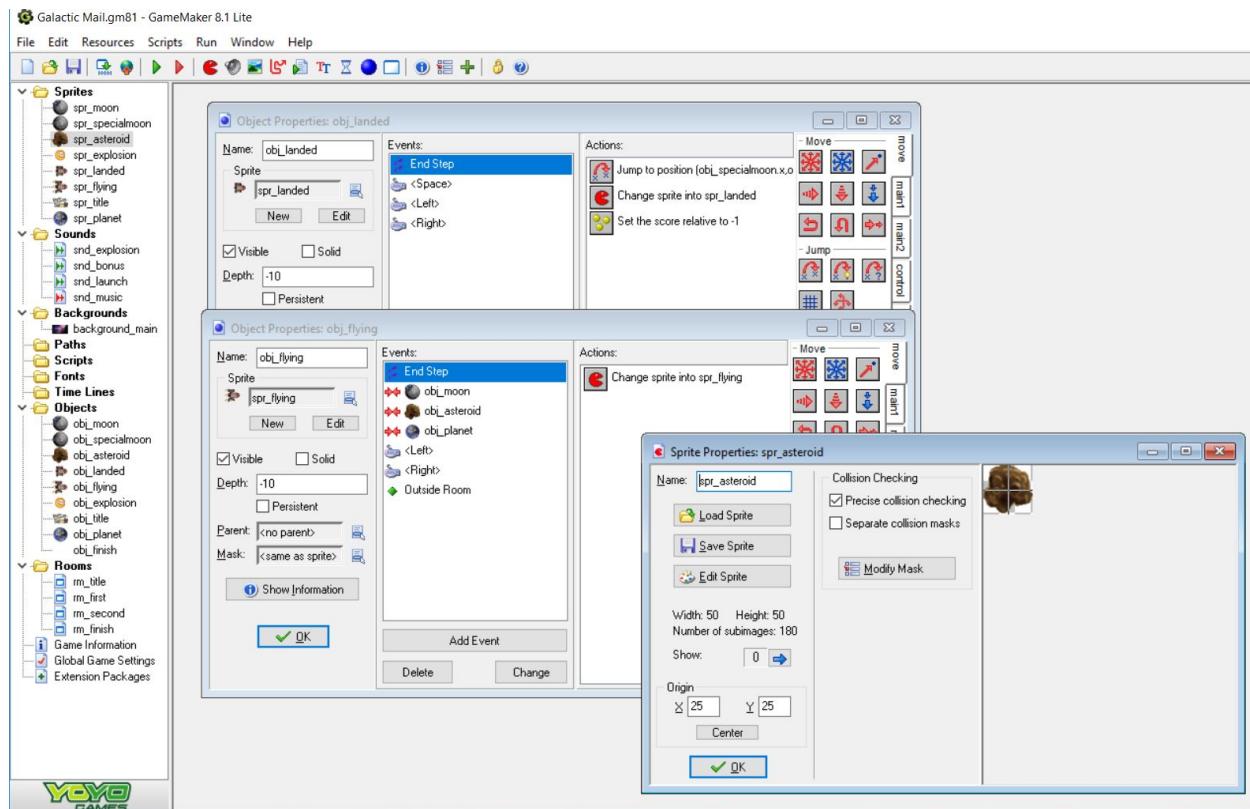


Fig 14: Galactic Mail developed in GameMaker using Drag and Drop coding

TicTacToe was also developed in GameMaker and the sprites were created in Microsoft Paint and Microsoft Word. The background and sound was obtained from free internet

sources. This game was coded using scripts in GameMaker language as opposed to the drag and drop features used in the Galactic Mail game (fig. 15)

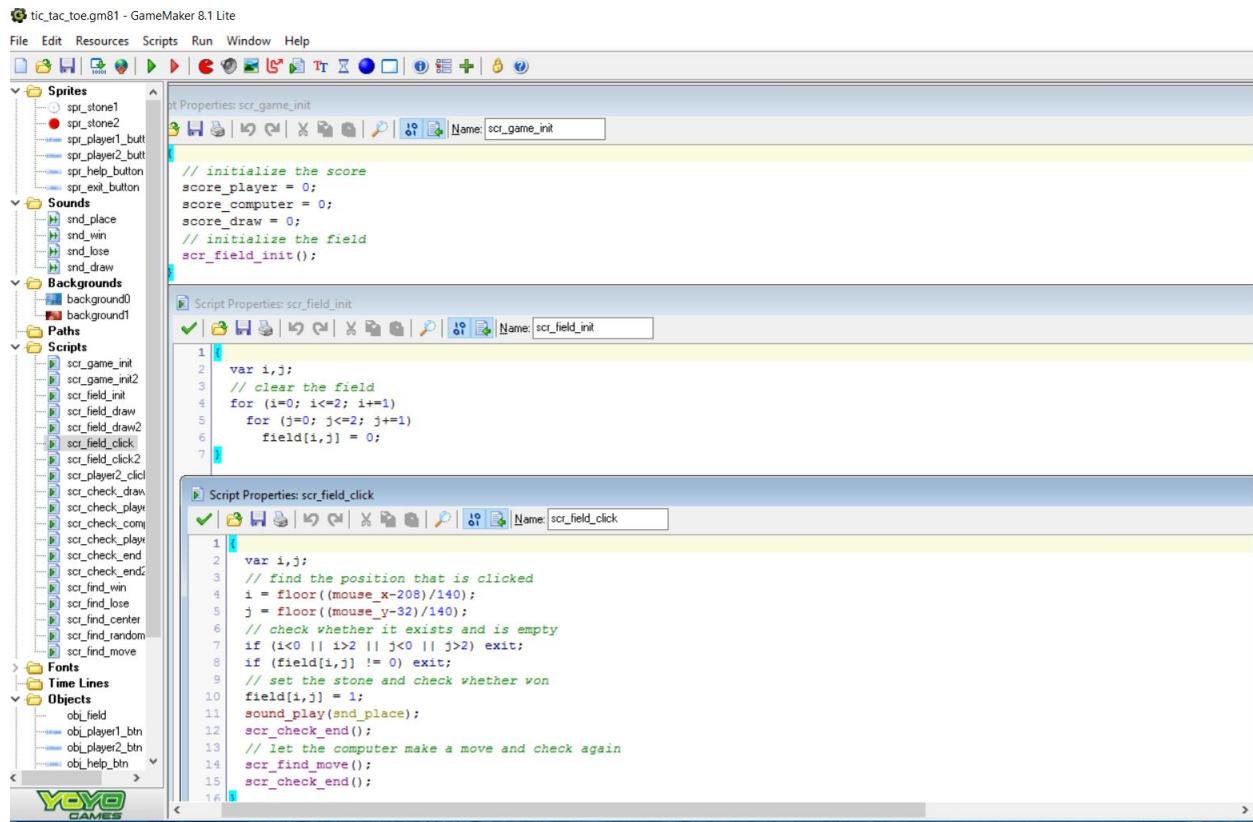


Fig 15: TicTacToe developed in GameMaker using scripts written in GameMaker language.

## Sprint Burndown Chart



Figure 16: Burndown chart for sprint 1

## Sprint 2

Timebox: 10days (February 12 - 22)

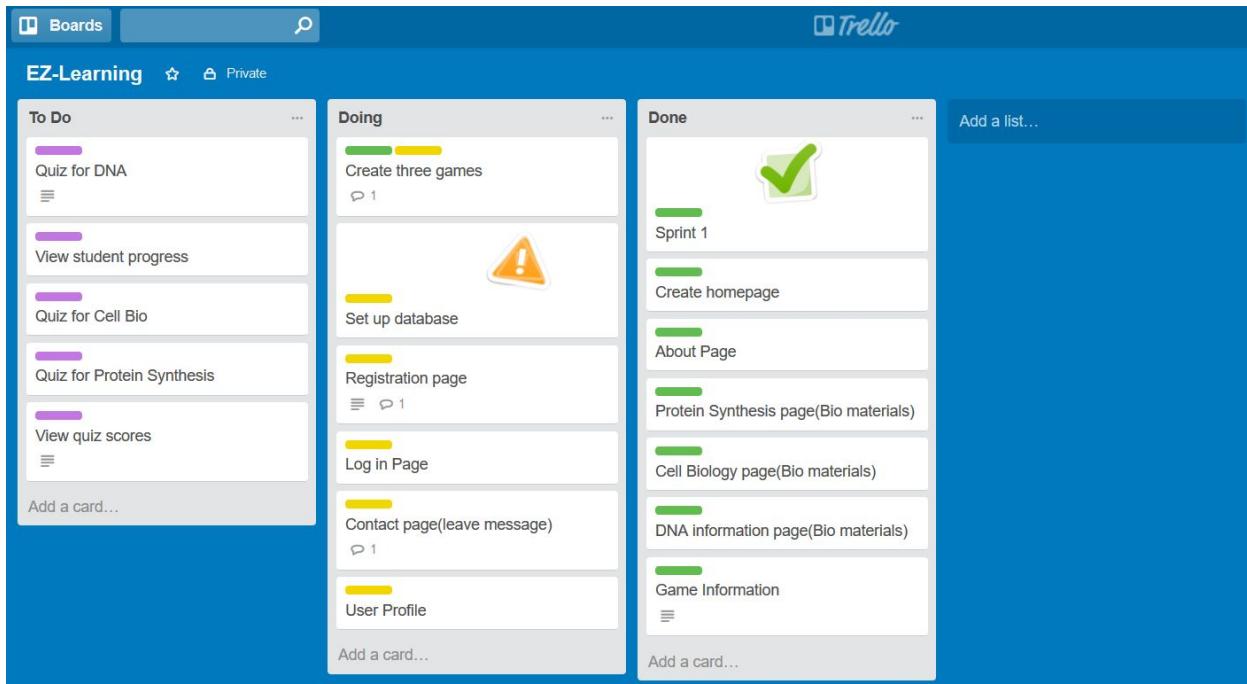


Fig 17: Kanban board for sprint 2 in Trello

## Backlog Refinement

The user stories(US) for this sprint were prioritized and the acceptance criteria for each US is defined. For sprint 2, US5 - US9 were implemented.

### US-5: Download Games

- Games are 100% functional
- Game links allow user to download an exe file
- Game link only appears when user score a 4 more greater on Quizzes

### US-6: Register for an Account

- All mandatory text field must be filled out
- Information is saved in the database
- User must have valid email account
- No duplication accounts
- Password must match
- Password must be 6 character long
- Registration page is consistent with other web pages

### US-7: Login to an Account

- Error prompts if password or username is incorrect
- Login status is shown
- Already logged in user cannot log in again unless user logout first
- Account is locked after 5 unsuccessful login attempts

#### US-8: Leave a Message

- Any user can use this feature located in the contact page
- Each steps is shown after the user clicks next
- A summary of the message is displayed at the end
- All mandatory field must be completed before form is submitted

#### US-9: View Profile

- User must be logged in, redirect user to login page if not logged in
- All mandatory field must be completed before form is submitted
- Information submitted is saved into the database
- Information is displayed with correct information when user is logged back in
- User may change password in profile page
- New password must have at least 6 characters or more

## Sprint Planning

The goals for sprint 2 is to finish the Virus Attack game from sprint 1 and create a database that can store user information. The first task is to complete the game since it was incomplete from the first sprint. In addition, create the registration process so that users can login to their account and perform tasks that requires membership such as having a profile page that would let users save general information or change their password.

### Sprint Backlog

#### US-5: Download Games

##### Tasks

- Development of Virus Attack game using GameMaker
- Integrate download link for games into webpage

#### US-6: Register for an Account

##### Tasks

- Design/implement page layout
- Utilize ASP.NET application services to manage membership, role, and profile management
- Utilize the Web Site Administration Tool (WSAT) to assign teacher or student roles to users
- Registration page is linked to login account page

- Validate user inputs

#### US-7: Login to an Account

##### Tasks

- Design/implement page layout
- Validate user inputs
- Utilize login controls to control what contents different users can view

#### US-8: Leave a Message

##### Tasks

- Design/implement page layout
- Utilize wizard control to create various steps such as basic user contacts, products, and a message box for user to leave comments
- Summarize information at the end
- Validate user inputs and show error message

#### US-9: View Profile

##### Tasks

- Design/implement page layout
- User can save profile information to database
- View quiz score page from profile
- Change password
- Redirect user to login page if not logged in

## Sprint Review Meeting

The first goal was to complete the third game which was incomplete from the first sprint due to time constraint. The main goal of this sprint is to set up the SQL server database to manage user information when they register for an account. When user have a registered account, they can login at anytime and also view their profile page. Any users can also leave a message in the contact page.

The third game called Virus Attack was similarly developed in GameMaker using the drag and drop feature. The idea for this game came from feedback received from reviewers of the first sprint who wanted more biology based games rather than science based games.



Fig. 18 Virus Attack game developed in GameMaker

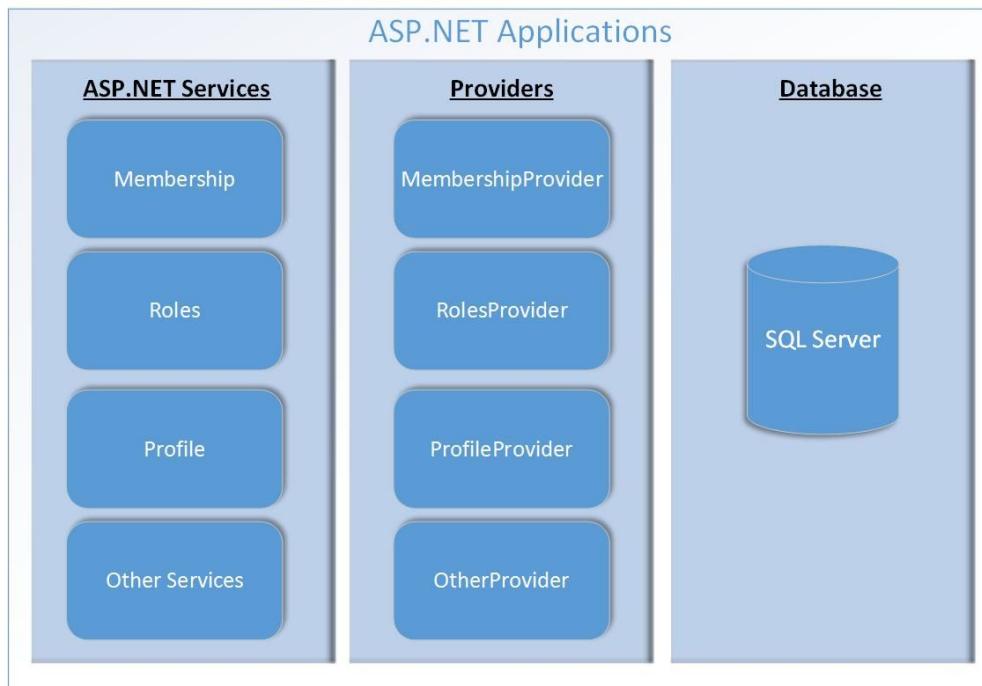


Fig 19: ASP.NET application services

The registration page was set up first using ASP.NET controls to talk to the configured application service providers and sets up the SQL database on the first time a user tries to register (fig 19). On the first attempt to register, the provider checks the connectionStringName attribute on the configured Membership provider in the web.config file and sets up the necessary tables in the database(fig 19).

```

<providers>
  <add name="DefaultProfileProvider" type="System.Web.Providers.DefaultProfileProvider, System.Web.Providers, Version=2.0.0.0, Culture=neutral, PublicKeyToken=31BF3856AD364E35" />
</providers>
</profile>
<membership defaultProvider="DefaultMembershipProvider">
  <providers>
    <add name="DefaultMembershipProvider" type="System.Web.Providers.DefaultMembershipProvider, System.Web.Providers, Version=2.0.0.0, Culture=neutral, PublicKeyToken=31BF3856AD364E35" />
  </providers>
</membership>

```

Fig.20: Membership provider in the web.config file

The registration process was created using the `CreateUserWizard` control asks the user to input a username, password, confirm password, and email to create an account(fig.20). These information are recorded into the Membership Table and the User Table inside the database(fig. 21).

Sign Up for your EZ-Learning Account

User Name:

Password:

Confirm Password:

E-mail:

**Create User**

Fig. 21 Registration form to sign up for an account

Update Script File: dbo.Memberships.sql

Name	Data Type	Allow Nulls	Default
UserId	uniqueidentifier	<input checked="" type="checkbox"/>	
ApplicationId	uniqueidentifier	<input checked="" type="checkbox"/>	
Password	nvarchar(128)	<input checked="" type="checkbox"/>	
PasswordFormat	int	<input checked="" type="checkbox"/>	
PasswordSalt	nvarchar(128)	<input checked="" type="checkbox"/>	
Email	nvarchar(256)	<input checked="" type="checkbox"/>	
PasswordQuestion	nvarchar(256)	<input checked="" type="checkbox"/>	
PasswordAnswer	nvarchar(128)	<input checked="" type="checkbox"/>	
IsApproved	bit	<input checked="" type="checkbox"/>	
IsLockedOut	bit	<input checked="" type="checkbox"/>	
CreateDate	datetime	<input checked="" type="checkbox"/>	
LastLoginDate	datetime	<input checked="" type="checkbox"/>	

Keys (1)  
PK\_dbo.Memberships (Primary Key, Clustered: UserId)

Check Constraints (0)

Indexes (2)  
IX\_ApplicationId (ApplicationId)  
IX\_UserId (UserId)

Foreign Keys (2)  
FK\_dbo.Memberships\_dbo.Applications\_ApplicationId (ApplicationId)  
FK\_dbo.Memberships\_dbo.Users\_UserId (UserId)

Triggers (0)

Fig.22 Membership table in SQL Server Database

The login page contains the Login control inside a LoginView control so that the login form is only shown to anonymous users. If a user is already logged in, the LoginView control display a message saying that the user is already logged in(fig.23). The login form also contain a link that will redirect users to the register page by clicking on “sign up now” text (fig. 23).

## Log in to EZ-Learning

The screenshot shows a login page with the following elements:

- A "Log In" button at the top right.
- A "User Name:" label and an input field.
- A "Password:" label and an input field.
- A "Remember me next time." checkbox.
- A "Log In" button at the bottom right.
- A "Sign Up Now" link.
- A "Login" link.
- A breadcrumb navigation bar at the top right showing "Home > Log in".
- A main title "Log in to EZ-Learning".
- A message "You are already logged in" followed by a "Logout" link.

```
<h2>Log in to EZ-Learning</h2>

<asp:LoginView ID="LoginView1" runat="server">
    <AnonymousTemplate>
        <asp:Login ID="Login1" runat="server" CreateUserText="Sign Up Now" CreateUserUrl="~/register.aspx"
            DestinationPageUrl="~/myProfile.aspx" >
        </asp:Login>
    </AnonymousTemplate>
    <LoggedInTemplate>
        You are already logged in
    </LoggedInTemplate>
</asp:LoginView>

<asp:loginstatus runat="server"></asp:loginstatus>
```

Fig. 23 The login page with Login controls

The user can login with a previously registered account and a login status is shown at the very top of the web page (fig. 24). This was created by using a LoginName control to display the name of the user but it is wrapped by the LoginView control so that anonymous users do not see it. In addition, the LoginStatus is also there to enable users to login and logout (fig 24). Since these codes are placed in the master page, it can be viewed in all the other pages so that the user will know when they are logged into the system.

The screenshot shows a master page with the following elements:

- A top navigation bar with "Home", "About", "Log in", and "Contacts".
- A "Logged in as student1" message followed by a "Logout" link.
- A central content area with a red and blue abstract background image.

```
<form id="form1" runat="server">
    <div id="right">
        <asp:LoginName ID="LoginName1" runat="server" FormatString="Logged in as {0}" />
        <asp:LoginView ID="LoginView1" runat="server">
            <LoggedInTemplate>
                <asp:LoginStatus ID="LoginStatus2" runat="server" />
            </LoggedInTemplate>
        </asp:LoginView>
    </div>
</form>
```

Fig. 24 Login controls

After registration, the user roles are assigned using Web Site Administration Tool (WSAT) (fig.25). Assigning roles to users limit user's access to certain pages that are protected. For instance, users with student as their role can access the quiz page to take a quiz while a user with a teacher role cannot access that page. The WSAT tool makes it easy to assign and manage roles as opposed to programmatically managing roles using a Role Manager API.

The screenshot shows the Microsoft ASP.NET Web Site Administration Tool (WSAT) interface. At the top, there is a navigation bar with links for Home, Security, Application, and Provider. Below the navigation bar, a message says "Click a row to select a user and then click **Edit user** to view or change the user's password or other properties. To as:". Underneath, another message says "To prevent a user from logging into your application but retain his or her information in your database, set the status".

**Search for Users**

Search by:  for:

Wildcard characters \* and ? are permitted.

[A](#) [B](#) [C](#) [D](#) [E](#) [F](#) [G](#) [H](#) [I](#) [J](#) [K](#) [L](#) [M](#) [N](#) [O](#) [P](#) [Q](#) [R](#) [S](#) [T](#) [U](#) [V](#) [W](#) [X](#) [Y](#) [Z](#) [All](#)

Active	User name			Roles
<input checked="" type="checkbox"/>	catd	<a href="#">Edit user</a>	<a href="#">Delete user</a>	<a href="#">Edit roles</a>
<input checked="" type="checkbox"/>	dg	<a href="#">Edit user</a>	<a href="#">Delete user</a>	<a href="#">Edit roles</a>
<input checked="" type="checkbox"/>	student1	<a href="#">Edit user</a>	<a href="#">Delete user</a>	<a href="#">Edit roles</a>
<input checked="" type="checkbox"/>	student2	<a href="#">Edit user</a>	<a href="#">Delete user</a>	<a href="#">Edit roles</a>
<input checked="" type="checkbox"/>	teacher1	<a href="#">Edit user</a>	<a href="#">Delete user</a>	<a href="#">Edit roles</a>
<input checked="" type="checkbox"/>	testnow	<a href="#">Edit user</a>	<a href="#">Delete user</a>	<a href="#">Edit roles</a>

[Create new user](#)

Add " **teacher1** " to roles:  
 Student  
 Teacher

Fig. 25 WSAT Tool to manage users

Once a user register for an account, the system will redirect the user to the default page which is the homepage. However, when a user logs into their account, they are automatically redirected to their profile page. The profile page can only be accessed when a user is logged in. If an anonymous user tries to access the profile page, they will be redirected to the login page. The profile page allows the user to save their general information such as first and last name, birth date, a short biography about them, and to change password. The profile properties can be added to the profile provider in the web.config file which can then be stored in the database (fig 26). Within the web.config is also a <location> attribute to prevent or authorize different types of users. For the profile page, the <location> attribute is used to deny users that are "?", this means that anonymous users are denied access to the profile page (fig 26).

```

<profile defaultProvider="DefaultProfileProvider">
  <properties>
    <add name="FirstName" />
    <add name="LastName" />
    <add name="DateOfBirth" type="System.DateTime" />
    <add name="Bio" />
  </properties>
  <providers>

```

```

<location path="myProfile.aspx">
  <system.web>
    <authorization>
      <deny users="?" />
    </authorization>
  </system.web>
</location>

```

Fig. 26 Profile codes inside web.config file

When the save button is clicked, the values entered in the textboxes are retrieved and stored in the profile. When the page is loaded a reverse of the process takes place so that the textboxes are pre-filled with values from the profile. However, to prevent the information from being lost the code gets the values from the profile only when the page initially loads and not during postbacks (fig 27).

```

public partial class myProfile : System.Web.UI.Page
{
  protected void Page_Load(object sender, EventArgs e)
  {
    if (!Page.IsPostBack)
    {
      FirstName.Text = Profile.FirstName;
      LastName.Text = Profile.LastName;
      DateOfBirth.Text = Profile.DateOfBirth.ToShortDateString();
      Bio.Text = Profile.Bio;
    }
  }

  protected void SaveButton_Click(object sender, EventArgs e)
  {
    if (Page.IsValid)
    {
      Profile.FirstName = FirstName.Text;
      Profile.LastName = LastName.Text;
      Profile.DateOfBirth = DateTime.Parse(DateOfBirth.Text);
      Profile.Bio = Bio.Text;
      lblUpdageMsg.Text = "Your Profile have been updated.";
    }
  }
}

```

Fig 27 Code behind in C# to store information provided by the user.

The user's information is stored inside the profile table inside the database which was passed by the profile provider when the user clicks on the save button (fig 28).

	UserId	PropertyName...	PropertyValue...	PropertyValue...	LastUpdate...
▷	9317f29a194d	FirstName:0...	catherineDo...	0x	5/5/2017 6:...
	60e3e7fe-1f...	FirstName:0...	Your first na...	0x	5/2/2017 10:...
🕒	af4510ae-fd...	FirstName:0...	catherineDo...	0x	5/10/2017 9:...
	NULL	NULL	NULL	NULL	NULL

Fig 28 Table data inside profile table of the database

The user can update their profile account by simple entering new information and clicking on the save profile button. Once completed, a message appears to notify the user that their profile has been updated (fig 29).

### My Account

Welcome to My Account page. You can personalize your profile in this page and change your password here.

First name

Last name

Date of birth

I am a college student.

Biography

Your Profile have been updated.

### View Quiz Scores

Change Your Password

Password:

New Password:

Confirm New Password:

Fig 29 Profile page layout with pre-filled in data entered by user

The contact page was the last feature to be implemented in this sprint. The contact page contains a wizard control that has multiple steps which allows the user to leave a message. The user have to fill out the mandatory field such as their first and last name, email, phone number, and contact method of preference (fig 30).

The screenshot shows a contact form wizard. On the left, a blue sidebar menu lists 'Contacts', 'Products', 'Comments', and 'Display'. The main area is titled 'Welcome to the Contacts Page'. It contains four text input fields for 'First Name', 'Last Name', 'Email', and 'Phone'. Below these are two radio buttons: 'Email' and 'Phone'. A 'Next' button is located at the bottom right of the form.

Fig 30 Contact page wizard with multiple steps

At the end of the contact form is a summary of the user's inputs in the previous steps (fig 31).

The screenshot shows a summary message box. It starts with 'Your Submission has been successfully completed and processed with the following:' followed by a list of user inputs: Full Name: Catherine Do, Email: sg@test.com, Phone: (333)111-8888, You prefer to be contacted by: Email, Interested in: Cloning, Comments: I need help with genetics.

Fig 31 Summary of message left by user

When the user clicks on the finish button on the last step, the values are saved and displayed in a summary text (fig. 32).

```

public partial class Controls_ContactForm : System.Web.UI.UserControl
{
    protected void Page_Load(object sender, EventArgs e)
    {
    }

    protected void Wizard1_FinishButtonClick2(object sender, WizardNavigationEventArgs e)
    {
        Result.Text = "Your Submission has been successfully completed and processed with the following:";  

        Result.Text += "<br />Full Name: " + Name.Text + " " + txtLastName.Text;  

        Result.Text += "<br />Email: " + txtEmail.Text;  

        Result.Text += "<br />Phone: " + txtPhone.Text;  

        Result.Text += "<br />You prefer to be contacted by: " + preference.SelectedValue;  

        Result.Text += "<br />Interested in: " + Products.SelectedValue;  

        Result.Text += "<br />Comments: " + txtComments.Text;
    }
}

```

Fig 32 the code behind to record the user's inputs

The contact form is created by using a wizard control and adding multiple steps to it. This form also includes user input validation as seen in fig 33.

```

<%@ Control Language="C#" AutoEventWireup="true" CodeFile="ContactForm.ascx.cs" Inherits="Controls_ContactForm" %>

<!--Wizard to create a contacts form--&gt;
&lt;center&gt;

    &lt;asp:Wizard ID="Wizard1" runat="server" ActiveStepIndex="0" BackColor="#EFF3FB"
        BorderColor="#B5C7DE" BorderWidth="1px" CellPadding="10" Font-Names="Verdana" Font-Size="Large"
        Height="292px" OnFinishButtonClick="Wizard1_FinishButtonClick2" Width="707px"&gt;
        &lt;HeaderStyle BackColor="#284E98" BorderColor="#EFF3FB" BorderStyle="Solid" BorderWidth="2px"
            Font-Bold="True" Font-Size="0.9em" ForeColor="White" HorizontalAlign="Center" /&gt;
        &lt;HeaderTemplate&gt;
            Welcome to the Contacts Page
        &lt;/HeaderTemplate&gt;
        &lt;NavigationButtonStyle BackColor="White" BorderColor="#507CD1" BorderStyle="Solid"
            BorderWidth="1px" Font-Names="Verdana" Font-Size="0.8em" ForeColor="#284E98" /&gt;
        &lt;SideBarButtonStyle BackColor="#507CD1" Font-Names="Verdana" ForeColor="White" /&gt;
        &lt;SideBarStyle BackColor="#507CD1" Font-Size="0.9em" VerticalAlign="Top" /&gt;
        &lt;StepStyle Font-Size="0.8em" ForeColor="#333333" /&gt;
        &lt;WizardSteps&gt;
            &lt;asp:WizardStep runat="server" title="Contacts"&gt;
                &lt;br /&gt;
                &amp;nbsp; First Name:&amp;nbs;
                &lt;asp:RequiredFieldValidator ID="RequiredFieldValidator6" runat="server"
                    ControlToValidate="Name" CssClass="ErrorMessage" ErrorMessage="Enter your first name"
                    TabIndex="1"&gt;&lt;/asp:RequiredFieldValidator&gt;
                &lt;asp:TextBox ID="Name" runat="server"&gt;&lt;/asp:TextBox&gt;
            &lt;/asp:WizardStep&gt;
        &lt;/WizardSteps&gt;
    &lt;/asp:Wizard&gt;
&lt;/center&gt;
</pre>

```

Fig 33 Codes from the wizard of the contact page

## Sprint Burndown Chart



Fig 34: Burndown chart for sprint 2

## Sprint 3

Timebox: 10days (February 28 - March 7)

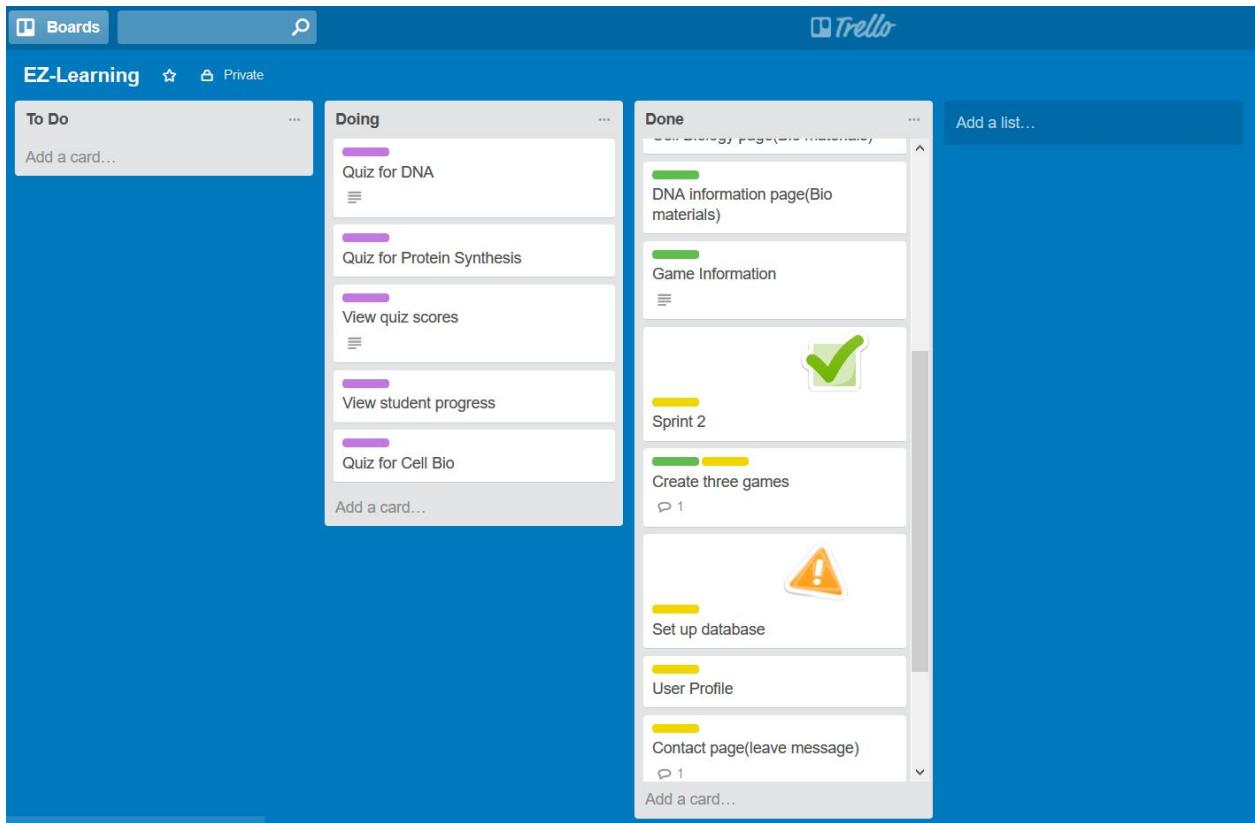


Fig 35: Trello board to show sprint 3 task cards in purple

## Backlog Refinement

The user stories(US) for this sprint were prioritized and the acceptance criteria for each US is defined. For sprint 3, US10 - US12 were implemented.

### US-10: Attempt Quiz

- Quiz have 5 questions with 5 answers
- Quiz scores are calculated at the end of the quiz
- Confirmation prompt required before start of quiz
- Redirect user to download page for a free game only if score is more than 4
- Quiz page is consistent with other web pages

### US-11: View Quiz Scores

- Scores are calculated and displayed correctly
- Scores are saved to the database to the correct user
- All student scores can be viewed by teacher users
- Quiz scores are based out of 5 total questions

### US-12: View Student Progress

- Only teacher users can view student scores
- Scores are displayed correctly and promptly

- All student scores are displayed
- Page is consistent with other web pages

## Sprint Planning

The goal for sprint 3 is to create three quizzes for each biology topic (DNA, Protein Synthesis, and Cell Biology). Each quizzes contain 5 questions that students have to answer. At the end of the quiz, students can submit the quiz and the system will calculate the score. If the student pass with a score of 4 or higher, then the system will redirect the user to a download page where the student can download a free game. If the user scored less than 4, the system will prompt the user to try again. The student's scores are saved in the database and students and teacher users may view it from a link in their profile.

### Sprint Backlog

#### US-10: Attempt Quiz

##### Tasks

- Create three quizzes with 5 questions each
- Only student users have authorization to attempt quiz
- Confirmation prompts user to confirm

#### US-11: View Quiz Scores

##### Tasks

- Link in profile to direct user to quiz score page
- Authorize student user access only
- Student quiz scores are displayed in a table
- Scores are stored and retrieved from database

#### US-12: View Student Progress

##### Tasks

- Link in profile to direct user to quiz score page
- Authorize teacher user access only
- All student scores are displayed in a table
- Scores are stored and retrieved from database

## Sprint Review Meeting

The main goal of sprint 3 was to create three quizzes that students can attempt and save those scores to a database that students and teacher users can view later on. Only student users can attempt to take the quizzes. Meanwhile, only teacher users can view every student's scores through a link on their profile page.

Students can attempt to take a quiz when they are ready. Once the students are finished with the quiz, the score is calculated and displayed. If the student passed with a score of 4 or higher, then the system would redirect the student to another page that contains a download link for the free game(Fig.36). If the student does not complete the quiz successfully than the system will display a message asking the student to try again.



#### DNA Quiz

Please fill in the 5 blank spaces with the correct answer. All answers must be in lowercase. (spelling counts!)

DNA does not unzip entirely. It unzips in a small area called a  fork. The four DNA bases are Thymine, Guanine, , and Adenine. Thymine pairs with . During DNA replication, an enzyme called  unwinds the double-stranded DNA. At the end of this process, DNA  seals up the fragments into one long continuous strand.

Congratulations! You score: 4 correctly out of 5. Your grade is: 80% Please wait. You will be given a downloadable link in 5 seconds.

Figure 36: Protein synthesis quiz showing that the user has passed with a score of 5/5 and will be redirected to another page to download a free game.

The user is redirected to another page with the game link in 5 seconds using System.Web.UI.HtmlControls (Fig.37).

```
lblResult.Text = "Congratulations! You score: " + correctAnswer.ToString() +  
    " correctly out of 5. Please wait. You will be given a downloadable link in 5 seconds.";  
HtmlMeta meta = new HtmlMeta();  
meta.HttpEquiv = "Refresh";  
meta.Content = "5;gameLink1.aspx";  
this.Page.Controls.Add(meta);
```

Figure 37: The codes to display the results and redirect the user to another page that gives the user a downloadable link for a free game after passing the quiz.

The correctAnswer variable is incremented every time the student got a question right. It is then stored in the score variable and inserted into the SQL Server database(Fig.38).

```
int score = correctAnswer;  
  
SqlDataSource userQuizDataSource = new SqlDataSource();  
userQuizDataSource.ConnectionString = ConfigurationManager.ConnectionStrings["ConnectionString"].ToString();  
userQuizDataSource.InsertCommand = "INSERT INTO [QuizScore] ([UserId], [Score], [DateTaken]) VALUES (@UserId, @Score, @DateTaken)";  
userQuizDataSource.InsertParameters.Add("UserID", Session["UserId"].ToString());  
userQuizDataSource.InsertParameters.Add("Score", score.ToString());  
userQuizDataSource.InsertParameters.Add("DateTaken", DateTime.Now.ToString());  
  
userQuizDataSource.Insert();
```

Figure 38:The codes to save the quiz score into the database.

The score is saved into the database and the student can view it through a link in their profile page (fig.39). The QuizScore data table contain the UserName, Score, DateTaken, UserID, and QuizTopic columns.

dbo.QuizScore [Data]				
	UserName	Score	DateTaken	UserId
	student1	5	5/9/2017 1:32:40 PM	4941f361-e...
	student1	5	5/9/2017 1:32:40 PM	af4510ae-fd...
	student2	4	5/11/2017 5:02:43 PM	cd59a8e7-b...
	NULL	NULL	NULL	NULL

Figure39: Database of quiz scores

The student can only view their quiz scores while teacher users can view all students quiz scores (fig.40). Users must be logged in to access their profile page and then click the view score button to access this page. The data is shown using a GridView control which selects the columns that I want to display from the SQL data source.

The screenshot shows two views of a quiz score database. The top view is for a student named 'student1'. The bottom view is for a teacher named 'teacher1'.

**Student View (Top):**

- Header: EZ-LEARNING
- Navigation: Home, About, Log In, Contacts
- User Status: Logged in as student1 [Logout](#)
- Table Data:

UserName	Score	DateTaken	Quiz Topic
student1	5	5/9/2017 1:32:40 PM	Protein Synthesis
student1	5	5/9/2017 1:48:26 PM	Cell Biology

**Teacher View (Bottom):**

- Header: EZ-LEARNING
- Navigation: Home, About, Log In, Contacts
- User Status: Logged in as teacher1 [Logout](#)
- Table Data:

UserName	Score	DateTaken	Quiz Topic
student2	4	5/11/2017 5:02:43 PM	DNA
student1	5	5/9/2017 1:32:40 PM	Protein Synthesis
student1	5	5/9/2017 1:48:26 PM	Cell Biology

- A note on the right side of the teacher view: "Teacher users can view all student's grades".

Figure 40: Teacher view vs student user view of quiz scores.

## Sprint Burndown Chart



Figure 41: Burndown chart for Sprint 3

## Test and Integration

Software testing is a crucial step in a Software Development Life Cycle. The purpose of testing is to check if the software meets the specified requirements by detecting faults related to functions and quality. Furthermore, testing can also determine the software quality by checking the quality attributes that are deemed important to the system.

## Objectives

The objective of testing the EZ-Learning web application is to improve on three major quality attributes: Usability, Security, and Performance. Usability is chosen because the goal of the EZ-Learning application is to make it easy for students to learn. Therefore, users ease of use for this application is of utmost importance. The quality attribute of security was chosen because the EZ-Learning application has to store sensitive student's grades. The EZ-Learning application's performance is also important because

the website needs to be responsive to increase user experience with the app so that users will want to use it.

Throughout the implementation of the EZ-Learning web application, for each sprint, the following tests were performed: unit testing, integration testing, system testing, and acceptance testing. These different levels of testing includes black box and white box testing.

## Approach

The testing approach includes both hands on testing and automated testing. Black box testing shall be used to test the function of the software to evaluate what the software does with invalid and valid inputs. Black box testing is conducted through the use of Visual Studio 2015 UI automation to simulate the way a user would use the application by a recording tool that writes out the codes. Manual black box testing of entering valid and invalid inputs shall be documented along with automated UI testing codes gathered from Visual Studio. In addition to black box testing, white box testing shall also be performed to ensure that a high quality product is produced. White box testing such as unit testing and integration testing shall be conducted on the system. Unit Tests shall be written on Visual Studio and analyzed.

## Test Tools

- Visual Studio 2015 Enterprise (UI Automation) to record the user clicks as they navigate through the web pages.
- Visual Studio 2015 for Unit Testing
- FireFox Developer Tools for performance testing

## Test Environment

### Hardware

Asus Laptop (Intel® Core™ i7 8.00GB RAM), running on Windows 10 on a 64bit operating system.

### Software

#### Visual Studio 2015

-automated tests through coded UI Tests (CUTs) to verify that the user interface is functioning correctly.

## Test Cases

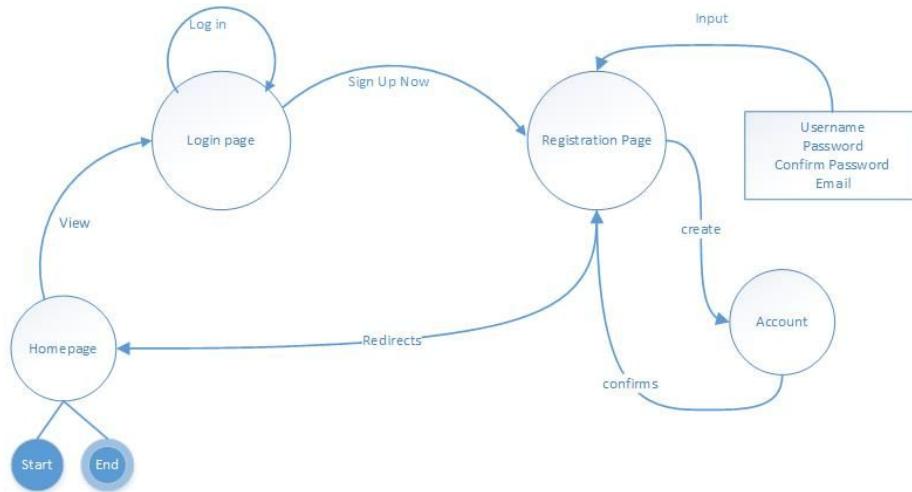


Figure 42: UI navigation diagram

<b>Test Case ID:</b>	TC-1
<b>Test Case Name:</b>	UI Testing of Registration process
<b>Description:</b>	User can register through the login page when they provide the necessary information need to create an account
<b>Created By:</b>	Catherine Do
<b>Date of Creation:</b>	March 7, 2017
<b>Executed By:</b>	Catherine Do
<b>Related Requirements</b>	NFR-2, FR-2
<b>Pre-conditions</b>	User must have a valid email
<b>Procedure</b>	<ol style="list-style-type: none"> <li>1. User access homepage</li> <li>2. Clicks on Login</li> <li>3. On the login page, user clicks on "sign up now"</li> <li>4. User fills out mandatory text fields</li> <li>5. Clicks on "create account"</li> <li>6. Confirmation message that account is created and user is redirected to homepage</li> </ol>
<b>Expected Result</b>	Account is created and stored in the database, user is then redirected to homepage

<b>Test Case ID:</b>	TC-2
<b>Test Case Name:</b>	Performance Load Test
<b>Description:</b>	Load testing of when a user logs in to their account
<b>Created By:</b>	Catherine Do
<b>Date of Creation:</b>	March 7, 2017
<b>Executed By:</b>	Catherine Do
<b>Related Requirements</b>	NFR-4, FR-1
<b>Pre-conditions</b>	User has a registered account
<b>Procedure</b>	<ol style="list-style-type: none"> <li>1. User access EZ-Learning website</li> <li>2. Clicks on Login tab</li> <li>3. Provide username and password</li> <li>4. Clicks Log in button</li> <li>5. User is redirected to profile page</li> </ol>
<b>Expected Result</b>	User logs into account and is redirected to their profile page with a loading time of less than 10 seconds

<b>Test Case ID:</b>	TC-3
<b>Test Case Name:</b>	Security test of valid and invalid inputs
<b>Description:</b>	The contact page is the target for this test, user inputs are manually tested.
<b>Created By:</b>	Catherine Do
<b>Date of Creation:</b>	March 9, 2017
<b>Executed By:</b>	Catherine Do
<b>Related Requirements</b>	NFR-1
<b>Pre-conditions</b>	N/A
<b>Procedure</b>	<ol style="list-style-type: none"> <li>1. User types in an incorrect web page address in the address bar</li> <li>2. User is redirected to an Error page</li> </ol>

<b>Expected Result</b>	Only valid inputs are allowed, invalid inputs should give errors
------------------------	--

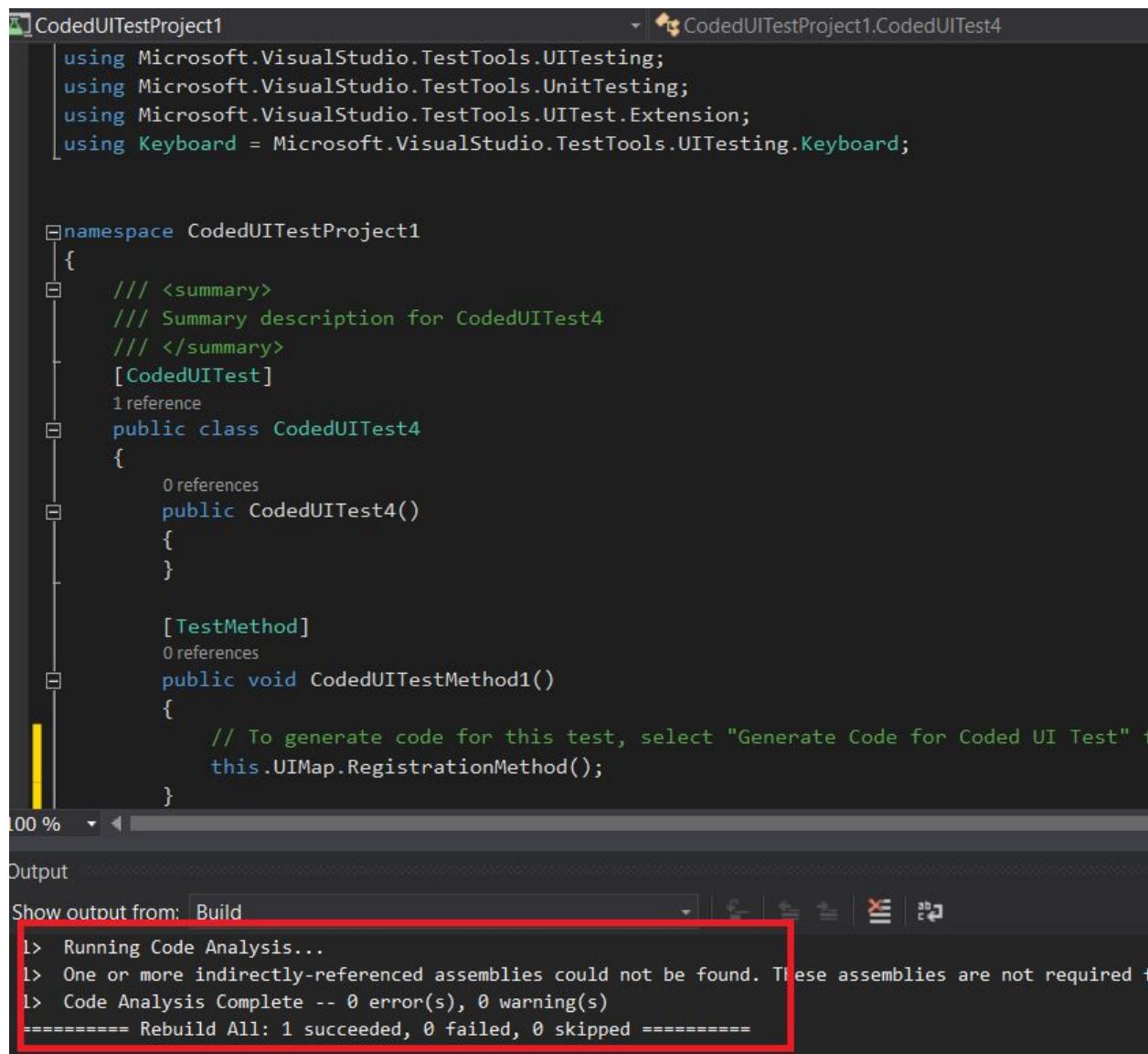
<b>Test Case ID:</b>	TC-4
<b>Test Case Name:</b>	Error handling
<b>Description:</b>	In case of unknown errors, webpage should still be available and show an error message that users would understand
<b>Created By:</b>	Catherine Do
<b>Date of Creation:</b>	March 13, 2017
<b>Executed By:</b>	Catherine Do
<b>Related Requirements</b>	NFR-3, FR-5
<b>Pre-conditions</b>	N/A
<b>Procedure</b>	<ol style="list-style-type: none"> <li>1. User access contact page</li> <li>2. User enter values into text boxes</li> <li>3. Clicks next when completed</li> <li>4. Next step is displayed if there are no errors</li> </ol>
<b>Expected Result</b>	User is redirected to an Error page in any unforeseen event

<b>Test Case ID:</b>	TC-5
<b>Test Case Name:</b>	Display Correct quiz score calculations
<b>Description:</b>	The percentage grade of the quiz score is calculated by dividing the number correct by the total number of questions and then multiplied by 100 to give the percentage.
<b>Created By:</b>	Catherine Do
<b>Date of Creation:</b>	March 21, 2017
<b>Executed By:</b>	Catherine Do
<b>Related Requirements</b>	N/A
<b>Pre-conditions</b>	N/A

<b>Procedure</b>	1. User logs in to student account 2. User attempts quiz 3. Clicks finished when quiz is completed 4. A message with the percentage score of the quiz is displayed
<b>Expected Result</b>	The quiz score is displayed as a percentage

## Test Results

### TC-1: UI Testing of Registration process



The screenshot shows the Microsoft Visual Studio IDE interface. The top navigation bar displays the project name "CodedUITestProject1" and the specific test file "CodedUITest4". The main area is a code editor with the following C# code:

```

using Microsoft.VisualStudio.TestTools.UITesting;
using Microsoft.VisualStudio.TestTools.UnitTesting;
using Microsoft.VisualStudio.TestTools.UITest.Extension;
using Keyboard = Microsoft.VisualStudio.TestTools.UITesting.Keyboard;

namespace CodedUITestProject1
{
    /// <summary>
    /// Summary description for CodedUITest4
    /// </summary>
    [CodedUITest]
    1 reference
    public class CodedUITest4
    {
        0 references
        public CodedUITest4()
        {

        }

        [TestMethod]
        0 references
        public void CodedUITestMethod1()
        {
            // To generate code for this test, select "Generate Code for Coded UI Test" on the context menu.
            this.UIMap.RegistrationMethod();
        }
    }
}

```

The bottom part of the interface is the "Output" window, which displays the results of a code analysis:

```

1> Running Code Analysis...
1> One or more indirectly-referenced assemblies could not be found. These assemblies are not required for this test.
1> Code Analysis Complete -- 0 error(s), 0 warning(s)
===== Rebuild All: 1 succeeded, 0 failed, 0 skipped =====

```

A red rectangular box highlights the output text from "Running Code Analysis..." to "===== Rebuild All: 1 succeeded, 0 failed, 0 skipped =====".

Figure 43: codes generated by the automated UI recorder

Using the automated UI recorder tool in Visual Studio 2015, I was able to record the UI navigation of registering for an account. I went through the process of signing up for an account and was able to successfully registered for an account so the test was recorded as a success.

## TC-2: Performance Load Test

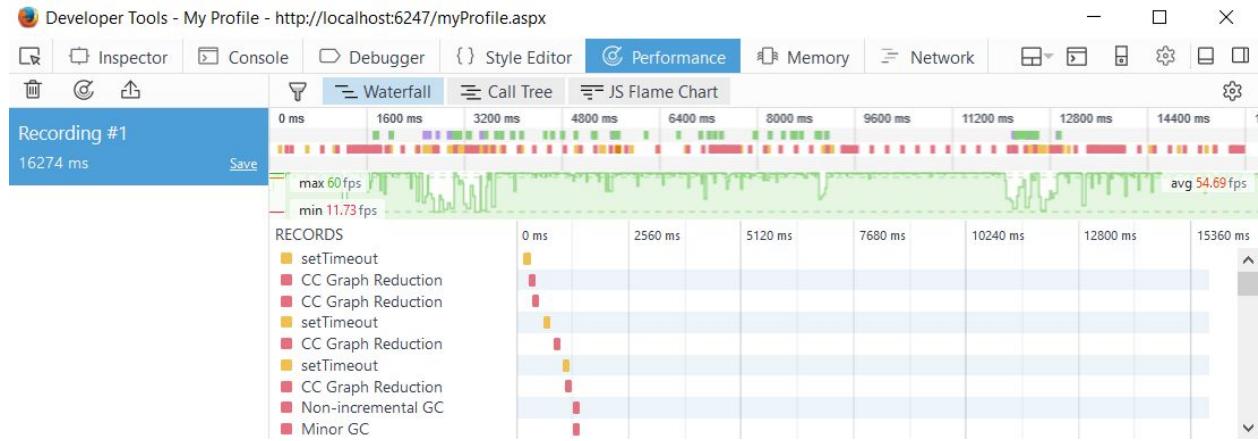


Figure 44: performance test generated by FireFox developer tools

The performance of the website was tested by using the developer tools in FireFox browser. In this test I was able to record the process of updating my account. I went through the homepage and logged into my account. Then I filled out my profile page and saved it. The whole attempt took about 16 seconds with an average of 54.69fps. The frame rate can be used to measure the responsiveness of a website [15]. The target frame rate for a smooth performance is 60fps. Therefore, it could be concluded that the EZ-Learning website passed the performance test with an average of 54.69fps.

### TC-3: Security test of valid and invalid inputs

The screenshot shows a contact form titled "Welcome to the Contacts Page". On the left, there's a sidebar with links: "Contacts", "Products", "Comments", and "Display". The main form has fields for First Name (catherine), Last Name (\*), Email (\*), and Phone (\*). Below the fields are radio buttons for "Email" and "Phone". A red-bordered box highlights an error message: "• Enter your last name  
• Enter a valid e-mail  
• Please enter a valid phone number". To the right of this box is a "Next" button.

Figure 45: Contact Form with error message for invalid inputs

The security of the website was tested on the contact page form. To protect my website from hackers, I used validation controls to protect from SQL injection attacks. I manually tested the text box field by entering random values. As expected, the system gave a system error to my invalid inputs. As a result, the security test is considered a passed for the contact page form.

### TC-4: Error Handling

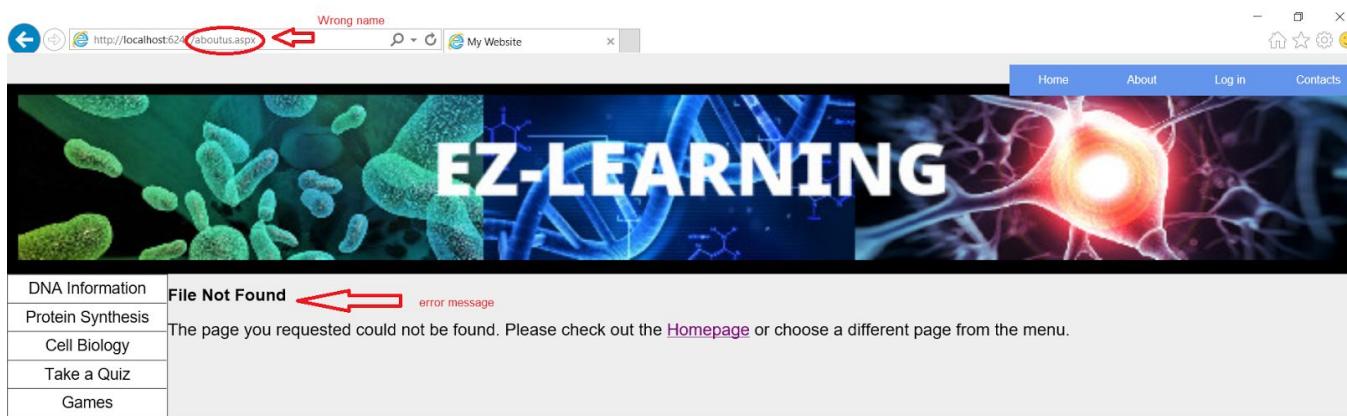
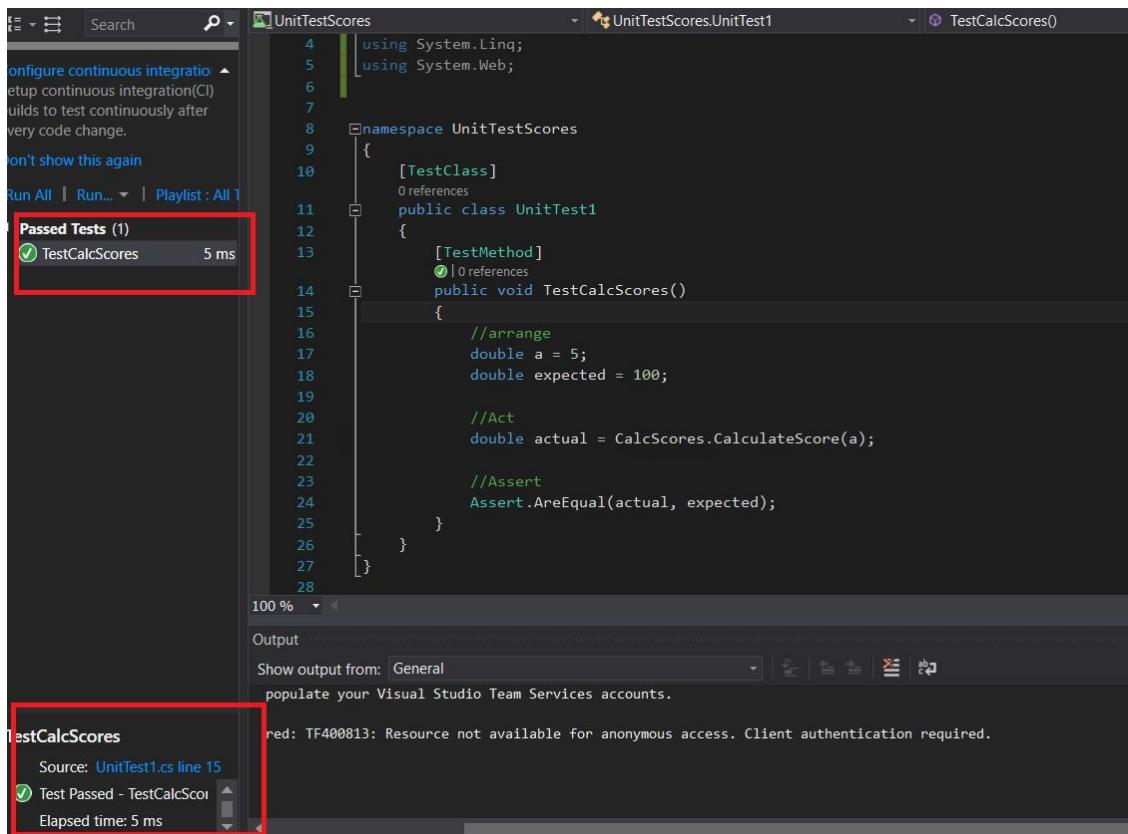


Figure 46: Error page

An important quality attribute for this application was availability. Therefore, I want to run tests that would create errors to see how the system would handle it. In the case that a user types in an incorrect address in the address bar, the system should redirect the

user to an error page instead of giving a coded error that users may not understand. Hence, the website will always be available even when there are errors.

## TC-5: Display Correct Quiz Score Calculations



The screenshot shows the Visual Studio 2015 IDE interface. On the left, the Test Explorer window displays a single passed test named "TestCalcScores" with a duration of 5 ms. This section is highlighted with a red box. On the right, the code editor shows a C# file named "UnitTest1.cs" under the namespace "UnitTestScores". The file contains a single test method, "TestCalcScores", which follows the classicArrange, act, assert pattern. The code uses Linq and System.Web namespaces. Below the code editor is the Output window, which shows a message about client authentication required. A second red box highlights the status bar at the bottom of the screen, which also displays the test name, source, and elapsed time.

```
4  using System.Linq;
5  using System.Web;
6
7
8  namespace UnitTestScores
9  {
10     [TestClass]
11     public class UnitTest1
12     {
13         [TestMethod]
14         public void TestCalcScores()
15         {
16             //arrange
17             double a = 5;
18             double expected = 100;
19
20             //Act
21             double actual = CalcScores.CalculateScore(a);
22
23             //Assert
24             Assert.AreEqual(actual, expected);
25         }
26     }
27 }
28
```

Passed Tests (1)  
TestCalcScores 5 ms

TestCalcScores  
Source: UnitTest1.cs line 15  
Test Passed - TestCalcScore  
Elapsed time: 5 ms

Figure 47: codes for unit testing

The unit testing for calculating the percent grade for the quiz score is tested in Visual Studio 2015. The test passed with an elapsed time of 5ms. The testing method used for this unit testing is the classic arrange, act, and assert pattern. All the necessary inputs and preconditions are arranged in the first step. Then for the act step, the actual work of the test is performed and finally, the expected result is verified in the assert step.

## Conclusion

The development of the EZ-Learning web application was to achieve two main objectives. The first is to improve on my programming skills and the second is to apply software engineering techniques and computer science principles learned in class. The EZ-Learning web application is a web based application that provide free

supplementary biology lessons for students. Its purpose is to assist teachers in bridging the gap in their interaction with students outside the classroom. This application makes it easy for students to learn biology anywhere and anytime. The EZ-Learning web application provides free games to motivate students to learn. Teachers can check student's progress by viewing their quiz scores. In developing this web application, I had a greater understanding of the software development lifecycle, from requirements gathering, designing the system, implementing, testing, and then deploying the system.

Agile Methodology was used in the development of this web application. The approach to the development of this project was a combination of Scrum and Kanban (Scrumban). Scrumban uses fixed length sprints from Scrum while in combination with Kanban to visualize the workflow and limit the work in progress [9]. A product backlog was created from using the Scrum approach and the web-based project management application, Trello was used to visualize the workflow for Kanban. This approach is excellent for small projects because it is lightweight and flexible to changes in requirements. The implementation of this application was split into three sprints. At the start of each sprint, I conducted a sprint planning event where I prepared the sprint backlog and commit to deliver the tasks for that sprint. During each sprint, I performed unit testing, integration testing, system testing, and acceptance testing. Since I performed various levels of White-Box testing and Black-Box testing during each sprint, I could detect errors at earlier stages of development. In the first and second sprint, I asked for feedback from friends and cohorts so that I could introduce new changes if necessary to the product backlog. Since I was using the Agile approach, it was not hard to make changes in the middle of the implementation stage.

Various technologies, programming languages, framework and libraries were used in the development of the EZ-Learning web application. The frontend was written in Javascript using ASP.NET framework. A jQuery library was used to create animation for validation controls in the UI. The technologies used were HTML5 and CSS. The backend was written in C# and a SQL server was used as the database management system(DBMS). The EZ-Learning web application used ASP.NET framework to create dynamic web pages which implements the MVC architecture design pattern. In addition, the games created for this web application was written in GameMaker Language(GML) which is built in the GameMaker application.

In completing the EZ-Learning web application, I deployed a functional web application by going through the entire software development lifecycle. I was able to utilize knowledge learned in class to develop this application. In the future, I would like to add more features to this application as I learn more new technologies. Through the success

of this project and with my current knowledge of the SDLC along with the different software methodologies, I hope to employ it future projects.

## Bibliography

- [1] Tapscott, D. 1998. Growing up digital: The rise of the Net Generation. New York: McGraw-Hill. Online [Assessed 2016]
- [2] Philip, D. "The Knowledge Building Paradigm: A Model of Learning for Net Generation Students". Online [Assessed 2016]
- [3] Becker, W. 2014. 25 STEM Education Apps You Need in 2015. Online [Assessed 2016]
- [4] Understanding Interactive Learning. Website [Assessed 2016]  
<http://www.scholastic.com/parents/resources/article/your-child-technology/understanding-interactive-learning>.
- [5] Ladas, C. Scrum-ban. Website [Assessed 2016]  
<http://leansoftwareengineering.com/ksse/scrum-ban/>
- [6] Pappas, C. How to Create a Successful Interactive eLearning Strategy. [Assessed 2016]  
<https://elearningindustry.com/7-tips-to-develop-successful-interactive-elearning-strategy>
- [7] Basic MVC architecture.  
[https://www.tutorialspoint.com/struts\\_2/basic\\_mvc\\_architecture.htm](https://www.tutorialspoint.com/struts_2/basic_mvc_architecture.htm). [Assessed 2017]
- [8] The editors of Encyclopaedia Britannica. Client-server architecture.  
<https://www.britannica.com/technology/client-server-architecture>. [Assessed 2017]
- [9] Atlassian. How the kanban methodology applies to software development.  
<https://www.atlassian.com/agile/kanban> [Assessed 2017]
- [10] Peterson David. What is Kanban?<http://kanbanblog.com/explained/> [Assessed 2017]

[11]Microsoft. What is Software Architecture?

<https://msdn.microsoft.com/en-us/library/ee658098.aspx> [Assessed 2017]

[12]Bass, Len, Clements, Paul, and Kazman, Rick. Software Architecture in Practice, 3rd Ed. SEI Series in Software Engineering, Addison-Wesley Professional, Oct. 2012.

[13]Atlassian. A brief look into using the scrum framework for software development.

<https://www.atlassian.com/agile/scrum> [Assessed 2017]

[14] Sabnis Mahesh. ASP.NET MVC 5: Using a Simple Repository Pattern for Performing Database Operations.

<http://www.dotnetcurry.com/aspnet-mvc/1155/aspnet-mvc-repository-pattern-perform-database-operations> [Assessed 2017]

[15] Zmi Wbammerg. Frame Rate. Mozilla Developer Network.

[https://developer.mozilla.org/en-US/docs/Tools/Performance/Frame\\_rate](https://developer.mozilla.org/en-US/docs/Tools/Performance/Frame_rate)

## Appendix A: Installation Instructions

### Download Source Code:

1. Download DoCatherine\_ProjectFinal zip file
2. Unzip the file using WinRar (downloadable at <http://www.rarlab.com/download.htm>)

### Download Visual Studio

1. Obtain Visual Studio from <https://www.visualstudio.com/downloads/>
2. Click on free Visual Studio download
3. Follow installation guide to install Visual Studio on PC

### Installing SQL Server

1. Go to <https://www.microsoft.com/en-us/sql-server/sql-server-downloads>
2. Search for the latest SQL Server Management Studio Express
3. Run the installer and follow the on-screen instructions

## Open WSAT

1. Start a command prompt by typing CMD on the Windows home screen and press enter.
2. Navigate to the folder where IIS is installed by entering: cd c:\Program Files\IIS Express
3. Enter the following command and hit enter:

Code:

```
iisexpress.exe/path:"C:\Windows\Microsoft.NET\Framework\v4.0.30319\ASP.NETWebAdminFiles" /vpath:"/ASP.NETWebAdminFiles" /port:8082 /clr:4.0 /ntlm
```

4. Start your browser and navigate to the following URL:

Code:

```
http://localhost:8082/asp.netwebadminfiles/default.aspx?applicationPhysicalPath=C:\Users\Cdo\Desktop\MyWebsite&applicationUrl=/
```

5. Sign in with window password and username
6. Click on Security tab
7. Under Users tab, click on Manage users
8. Assign User roles

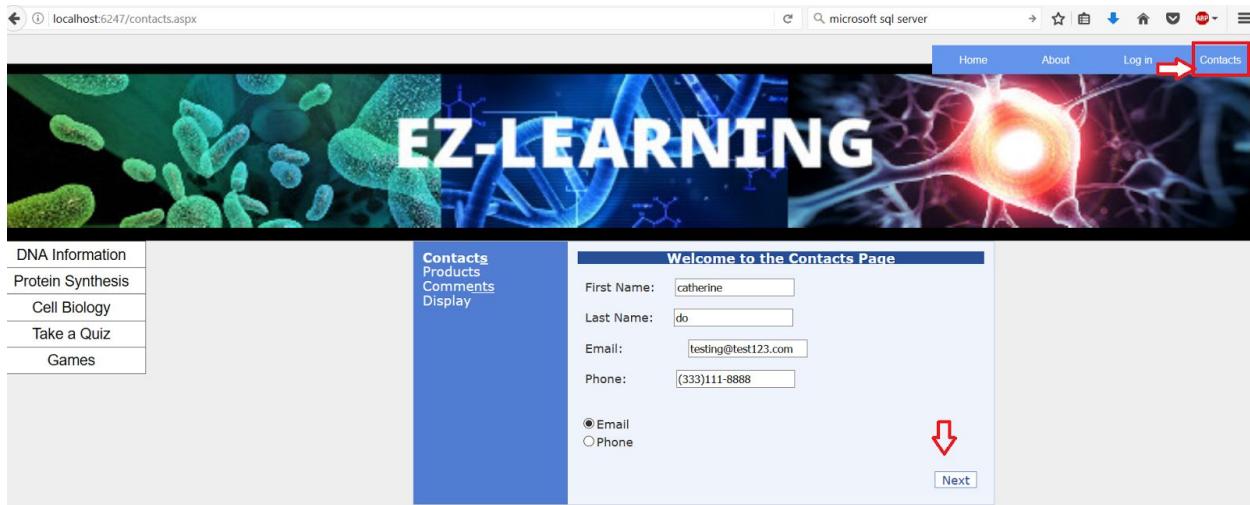
## Run Application

1. Open Visual Studio
2. Go to File -> open -> website
3. Locate downloaded file DoCatherine\_ProjectFinal
4. Double click on file and open MyWebsite folder
5. Press the green run button and choose a browser to run the website

## Appendix B: Operational Manual

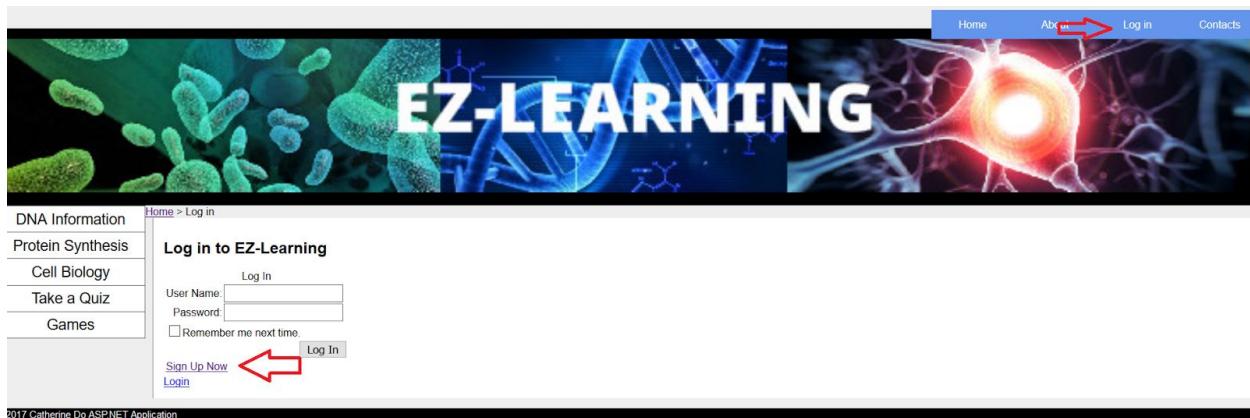
This section will show the user how to operate the EZ-Learning Web Application through images and short written instructions.

## How to use the contact page



Navigate to the contact page by following the red arrow at the very top right corner of the page. Enter the necessary field on the contact form and click “next”. Fill out the next few steps and click “submit”. A summary of the information entered in the contact form will appear on a successful submission.

## How to Register/Login



Navigate to the log in page by clicking the login tab at the very top-right corner of the website. On the log in form, the user can input their username and password if the user already have an account. Otherwise, the user can click on the “sign up now” link to be redirected to the registration page.



Sign Up for your EZ-Learning Account

User Name:

Password:

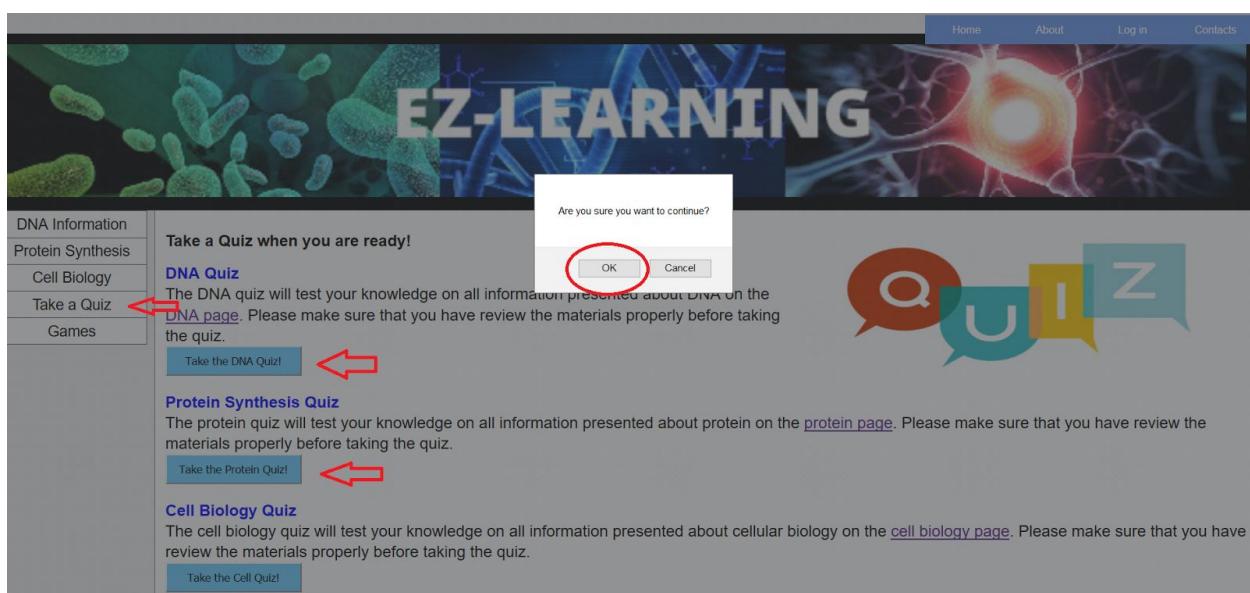
Confirm Password:

E-mail:

**Create User**

At the registration page, fill in the blank text fields with valid information and then click on “create user”. If registration is successful, there will be a confirmation message and the user is redirected to the homepage. Otherwise, an error message will display and the user must fix the error to complete registration.

## How to take the Quiz



The screenshot shows the EZ-Learning website. At the top, there's a navigation bar with links for Home, About, Log in, and Contacts. The main content area has a banner with the text "EZ-LEARNING". On the left, a sidebar menu lists DNA Information, Protein Synthesis, Cell Biology, Take a Quiz (which is highlighted with a red arrow), and Games. The "Take a Quiz" section contains three items: DNA Quiz, Protein Synthesis Quiz, and Cell Biology Quiz, each with a "Take the [Quiz]!" button. A red arrow points to the "Take the DNA Quiz!" button. In the center, a modal dialog box asks "Are you sure you want to continue?" with OK and Cancel buttons. To the right, there's a decorative graphic of speech bubbles with the word "QUIZ".

Navigate to the “take a quiz” tab on the left side of the web page. At the quiz page, there are three biology topics to choose from. Pick one topic and click on the button to take the quiz. Click on the “Ok” button to confirm. If the user is not logged in to a student account, the user will be redirected to an error page that displays an error message telling the user to log in to a student account. Otherwise, the user is redirected to the quiz page to take the quiz.

Logged in as student11 [Logout](#)

Home About Log in Contacts

# EZ-LEARNING

DNA Quiz

Please fill in the 5 blank spaces with the correct answer. All answers must be in lowercase. (spelling counts!)

DNA does not unzip entirely. It unzips in a small area called a [ ] fork. The four DNA bases are Thymine, Guanine, [ ], and Adenine. Thymine pairs with [ ]. During DNA replication, an enzyme called [ ] unwinds the double-stranded DNA. At the end of this process, DNA [ ] seals up the fragments into one long continuous strand.

[Submit](#)

The user must fill out all of the text field and then click “submit” once finished. The system will grade the quiz and display the scores.

The screenshot shows the EZ-LEARNING website. On the left, there is a vertical sidebar with links: DNA Information, Protein Synthesis, Cell Biology, Take a Quiz, and Games. The main content area displays a DNA quiz with a question about DNA replication. A modal dialog box is overlaid on the page, titled "Opening tic\_tac\_toe.exe". The dialog contains the following text:  
You have chosen to open:  
 **tic\_tac\_toe.exe**  
which is: Binary File (5.3 MB)  
from: http://localhost:6247  
Would you like to save this file?  
[Save File](#) [Cancel](#)

If the user scores higher than 4, then the system will redirect the user to the game page where the user can download a free game.

## How to view/update user's Profile

The screenshot shows a user interface for managing a profile. At the top, there is a decorative banner with various scientific and medical icons. On the left, a vertical navigation bar lists several tabs: 'Information', 'Synthesis', 'Biology', 'Quiz', and 'Notes'. The 'Quiz' tab is currently active, indicated by a grey background. The main content area has a title 'My Account' and a welcome message: 'Welcome to My Account page. You can personalize your profile in this page and change your password here.' Below this, there are input fields for 'First name' (Kat), 'Last name' (Do), and 'Date of birth' (11/11/1999). A text area labeled 'Biography' contains the text 'I am a college student.' A red arrow points from the text 'Your Profile have been updated.' back to the 'Save Profile' button. Below this section, there is a heading 'View Quiz Scores' with a 'View Quiz Scores' button. Underneath it is a 'Change Your Password' section with fields for 'Password', 'New Password', and 'Confirm New Password', along with 'Change Password' and 'Cancel' buttons.

User must be logged in first to access the profile page. When a user first logged in, the user is redirected to the profile page immediately. However, the user can also access the profile page by hovering over the login tab at the navigation bar and clicking on the "My account" tab. The user can update their profile by filling in the required text field and clicking on the "save profile" button. User may also change their password and view their quiz scores through the profile page.