Online Interactive Mathematical Visualizations

Adam Zess

Sen Yan

**Final Report**

Table Of Contents

**Conceptual Operation (CONOPS) 4**

[**Executive Summary**](#_tyjcwt) **8**

[**Introduction**](#_mevfvd3ig4w8) **9**

[2.1. Background](#_1t3h5sf) 9

[2.2. Overview](#_4d34og8) 9

[2.3. Referenced Documents and Standards](#_2s8eyo1) 10

[[2]Kay, R. H. (2011). Evaluating learning, design, and engagement in web-based learning tools (WBLTs): The WBLT Evaluation Scale. Computers in Human Behaviour, 27, 1849–1856.](#_cat217knrqtu) 10

[**Operating Concept**](#_3rdcrjn) **11**

[3.1. Scope](#_26in1rg) 11

[3.2. Operational Description and Constraints](#_lnxbz9) 11

[3.3. System Description](#_35nkun2) 11

[3.4. Modes of Operations](#_1ksv4uv) 12

[3.5. Users](#_44sinio) 12

[3.6. Support](#_2jxsxqh) 12

[**4. Scenario(s)**](#_z337ya) **13**

[4.1. Student Login](#_axxgdyyfwa7e) 13

[**5. Analysis**](#_1y810tw) **13**

[5.1. Summary of Proposed Improvements](#_4d3dbswv6wfm) 13

[5.2. Disadvantages and Limitations](#_2xcytpi) 13

[5.3. Alternatives](#_1ci93xb) 14

**Functional System Requirements 14**

[**1. Introduction**](#_gjdgxs) **18**

[1.1. Purpose and Scope](#_30j0zll) 18

[2.1. Applicable Documents](#_1fob9te) 18

[2.2. Reference Documents](#_3znysh7) 19

[3.1. System Definition](#_2et92p0) 20

[3.2.1. Functional / Performance Requirements](#_3xhg0t7yii8v) 21

[3.2.1.1. Student Web Navigation](#_3tws5kj9c28z) 21

[3.2.1.2. Exam Module Access](#_5eqwa5x245oy) 21

[3.2.1.3. Survey Responses and Review](#_u8lzsktg0jvg) 21

[3.2.1.4. Administrator Dashboard](#_1ab8zg3j2snd) 21

[3.2.1.5. Administrator Content Implementation](#_19abl0pyfifw) 21

[3.2.2. Outputs](#_3j2qqm3) 21

[3.2.2.1 Student Interface Website Structure](#_2ihjcond0awy) 21

[3.2.2.2 Examination Results](#_vguiwfw6xdya) 21

[3.2.2.3 Student Visualization Activity](#_30foo3bfko4k) 22

[**4. Support Requirements**](#_49x2ik5) **22**

[**Appendix A Acronyms and Abbreviations**](#_3o7alnk) **22**

**Interface Control Document (ICD) 22**

[**Overview**](#_3dy6vkm) **26**

[**References and Definitions**](#_z0rwqnr76ntz) **27**

[References](#_xeies9nscmm) 27

[Definitions](#_mpv8ja2e5uyt) 27

[**Software Interface**](#_kf8t3f2sflcv) **28**

[Student-User Interface](#_wjyo3i3mr2ly) 28

[SiteGround Web Service](#_egajpqalsfw1) 28

[WordPress Content Management System](#_rhycmriyb595) 28

[Administrator and Exam Interface](#_is31n8im54xk) 30

[Exam Server](#_ba0ko23puqr7) 30

[Exam and Visualization Implementation](#_9ylmb3ycdb6b) 30

Appendix C - Subsystem Reports 32

Student-User Interface Subsystem Reports 33

1. Subsystem Introduction 34
2. Subsystem Overview 34
   1. SiteGround Web Hosting Service 34
   2. WordPress Content Management System 35
3. Data Collected 36
4. Validation Plan 37

Administrator Interface Subsystem Reports 39

1. Subsystem Introduction 40
2. Subsystem Overview 40
3. Data Collected 41
4. Validation Plan 42

SYSTEM DESCRIPTION AND DEVELOPMENT 43

1. Overview 44
2. Development Plan and Execution 44
   1. 2.1 Plan 44
   2. 2.2 Data 45
   3. 2.3 Conclusion 46

Appendix D - Administrator Interface Source Code 48

Online Interactive Mathematical Visualizations

Adam Zess

Sen Yang

**Concept of Operations**

Concept of Operations

for

Online Interactive Mathematical Visualizations

Team<Mathematics SW>

Approved by:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Project Leader Date

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Prof. Kalafatis Date

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

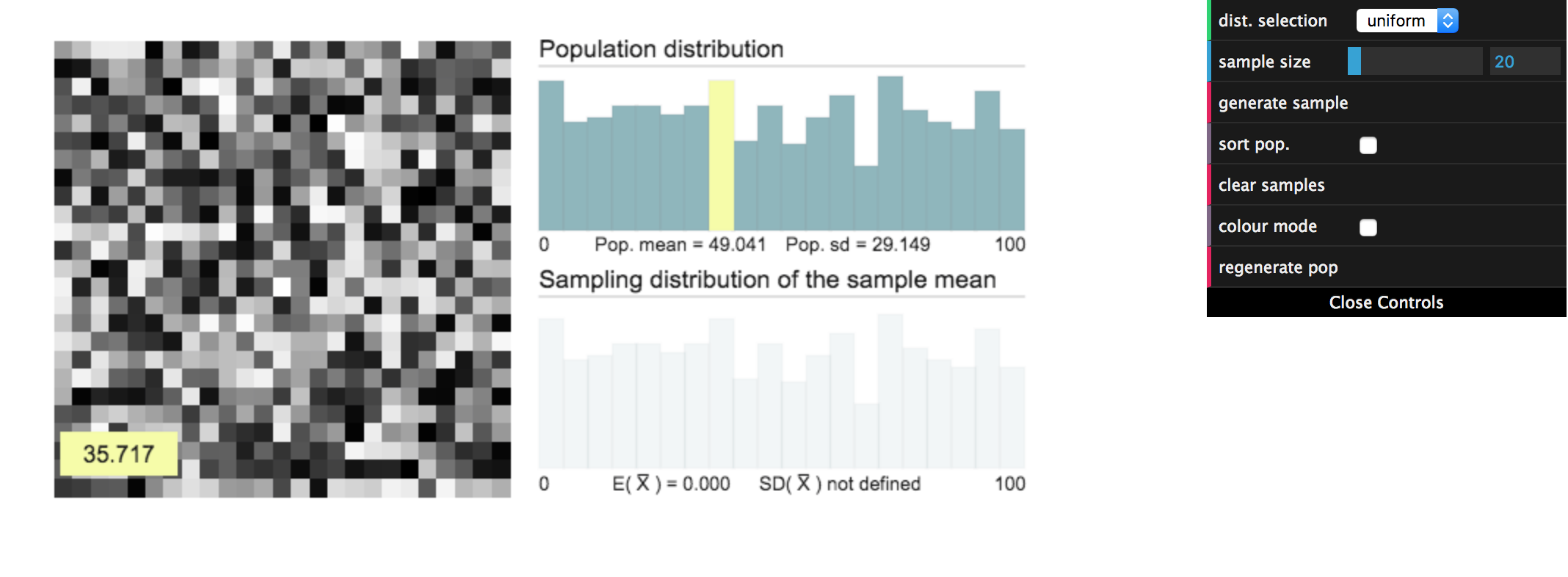
T/A Date

**Change Record**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Rev.** | **Date** | **Originator** | **Approvals** | **Description** |
| **1** | 2/5/2018 | Sen Yang |  | Draft Release |

# **Executive Summary**

In this project, we are designing a mathematical learning website for Dr. Jim Pettigrew of Western Sydney University. This project aims to address the problem of the mathematical under-preparedness of students transitioning into university study, and the risk this poses to their engagement. It seeks to improve students’ engagement with their mathematical studies by allowing them to collaboratively explore visual representations of important concepts in the discipline. The use and integration of Online Interactive Mathematical Visualizations (OIMVs) to the website with the goal to enhance student learning a crucial component to this project.



**Figure 1. Conceptual Image of OIMVs**

# **Introduction**

Mathematics has held an established position among the humanities for more than 2,000 years. Mathematical knowledge, with old questions being answered and new ones asked, is growing at an unprecedented rate. With the development of the mathematical science, people need more and more advanced technology to explore the study of mathematics.

In the information technology era, web designers are trying to build scientific and aesthetically pleasing websites for users. In this project, we will develop a website for the College of Mathematics of Western Sydney University in Australia. The purpose of the design is to import a variety of visualization elements to the slides and make lessons more attractive to improve students’ engagement with their mathematical studies.

## ***2.1. Background***

In early 2014, researchers at the University of Western Sydney developed a suite of interactive mathematical visualisations (IMVs) aimed at improving students’ understanding of key concepts in first-level mathematics. The pedagogical and interaction design considerations that formed this development, as well as the processes adopted in preparing the IMVs for classroom use. Guidelines for interaction design that draw on research in human-computer interaction, information visualisation and cognitive technologies are reviewed and contextualised for the specific learning needs of mathematics students at UWS. Examples are given of how these guidelines were factored-in to the IMV development. In addition to the pedagogical and technological dimensions of this work, research-based methods for analysing and evaluating the educational effectiveness of the IMVs are examined. It is expected that these methods will underpin a formative evaluation approach to ongoing design and development of the IMVs.

## ***2.2. Overview***

Our website is designed for mathematical courses with an online interactive mathematical visualization requirement and with the goal to improve students’ engagement with mathematics studies. A further aim is to pilot the use of these visualisation tools as examples of an approach to curriculum that combines dynamic online content with opportunities for socially constructed learning and formative assessment.

*2.2.1. Interface System*

HTML, CSS, jQuery, Bootstrap and JavaScript will be utilized to complete the front-end design of the website.

*2.2.2. Database System*

PHP, cookie, Session, HTTP server (Apache ) and MySQL will be utilized to complete the back-end design of the website.

*2.2.3. Interaction between front-end and back-end.*

The website will be developed using the following process and will meet the requirements outlined below:

Ajax can be used to complete the data interaction between front-end and back-end. Ajax can request for and connect data without refreshing the entire webpage. Json is a specific format which describes data interaction between the front-end and back-end of the website to analyze data. When the front-end sends a request (Ajax), the back-end can analyze the field, and obtain the data from the database (MySQL) and transfer the data to Json format for the front-end to interpret. The front-end can accept the data (Ajax) and analyze the data (Json) to complete the interaction between the two ends of the website.

## ***2.3. Referenced Documents and Standards***

[1]Asiala, M., Cottrill, J., Dubinsky, E., & Schwingendorf, K. E. (1997). The development of students’ graphical understanding of the derivative. The Journal of Mathematical Behavior, 16(4), 399–431.

## [2]Kay, R. H. (2011). Evaluating learning, design, and engagement in web-based learning tools (WBLTs): The WBLT Evaluation Scale. Computers in Human Behaviour, 27, 1849–1856.

[3]Sedig, K. (2009). Interactive Mathematical Visualizations: Frameworks, Tools, and Studies. In E. ZudilovaSeinstra, T. Adriaansen, & R. van Liere (Eds.), Trends in Interactive Visualization (pp. 343–363). SpringerVerlag

# **Operating Concept**

## ***3.1. Scope***

In this project, we are designing the website for college of mathematics for internal use. Students are able to review slides, attempt quizzes, and check their grades. The faculty is able to track students’ learning progress in real-time. The home-page and subpages are concise with only necessary functions. The website will be completed by the end of March, 2018. The budget of the project will be within $1,000.

## ***3.2. Operational Description and Constraints***

The point of designing the web page is to define where components of the webpage appear on screen, but we can do so only relative to the user’s viewing environment. People see websites’ pages using various viewpoints with different monitor sizes and resolutions. The range of sizes of screens that we will be developing our website to be compatible for is from smartphone sizes to wall-mounted monitor sizes. Another constraint of website design is that unlike printed designs, where the viewing area of any design is fixed, web users can (and do) zoom in or out as they interact with a web page, changing the size of text and images. Different browsing environments handle zoom differently — some enlarge images as text is enlarged, and other times enlarging text doesn’t affect other page elements. The Online Interactive Mathematical Visualizations website will need to be developed in a way that allows zoom functionality effectively.

## ***3.3. System Description***

The system will be developed into two distinct subsystems: the front-end system and the back-end system. The front-end system is primarily used to send queries and requests, and receive data from the back-end system or the host information system. It serves or provides users with the ability to interact and use an information system. Typically, front-end systems have very limited computational or business logic processing capabilities and rely on the data and functions from the host system. However, some advanced level front-end systems do maintain copies of data, such as a duplicates of each transaction sent to the back-end system. A front-end system includes or consists of textual or graphical user interface (GUI) and/or a front-end client application that is connected by the back-end system.

The separation of front-end and back-end computer systems simplifies the computing process when dealing with multilayered development and maintenance. Back-end systems deal with databases and data processing components, so the purpose of the back-end system is to launch the operating system's programs in response to front-end system requests and operations. In other words, the back-end system implements responses to what the front end has initiated.

## ***3.4. Modes of Operations***

There are three distinct functionalities that can be utilized on the OIMVs webpage:

1. Review lessons: User will review assigned slides and prepare for quizzes.
2. Take quizzes: The user will take quizzes to engage in self-check of studies.
3. Check grades: The user should be able to check their grades on the quizzes.

## ***3.5. Users***

Users are college students who register for the mathematical courses that are implementing the use of the online interactive mathematical visualization course website and the faculty members who take responsibility for teaching the corresponding courses.

## ***3.6. Support***

The online interactive mathematical visualization system is designed for the College of Mathematics at Western Sydney University in Australia.

# **4. Scenario(s)**

## ***4.1. Student Login***

* + Delivered mathematical learning material in the form of micro-lessons.
  + Micro-lessons to include pre- and post-quizzes, as well as feedback-rich prompts to check students understanding of lessons.
  + Interactive mathematical visualizations to supplement lessons.

***4.2. Faculty Login***

* Instructor-monitored database creation to check student’s interactions and grades.

# **5. Analysis**

## ***5.1. Summary of Proposed Improvements***

* Mathematical learning material – delivered as micro lessons –dedicated to a particular mathematical concept (e.g. the limit definition of the derivative, the delta-epsilon definition of the limit or the Central Limit Theorem).
* A pair of (pre and post) quizzes, as well as feedback-rich prompts to allow students to check their understanding as they progress through the site.
* A focus on lessons rich in interactive mathematical visualizations.
* A database recording student’s identifying information, their interactive behavior with the website (particularly the visualization at its center), performance on the pre and post quizzes and self-checking prompts.
* An instructor’s dashboard that would allow real-time monitoring of the information recorded in the database.

## ***5.2. Disadvantages and Limitations***

Some of the disadvantages of the OIMVs are:

* The webpage may not perform as well on mobile devices as it will on a desktop or laptop.
* The new interface might not be aesthetically satisfying to every student who uses the website
* Students may need some time to adapt the new system.

## ***5.3. Alternatives***

The proposed project has been outlined specifically and will be implemented to satisfy the specific requirements described above - there are no alternatives that we are currently considering.

Online Interactive Mathematical Visualizations

Adam Zess

Sen Yang

**Functional System Requirements**

REVISION – Draft

27 April 2018

Functional System Requirements

for

Online Interactive Mathematical Visualizations

Prepared by:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Mathematics SW Team Date

Approved by:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Project Leader Date

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Prof. S. Kalafatis Date

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

T/A Date

**Change Record**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Rev.** | **Date** | **Originator** | **Approvals** | **Description** |
| **1** | Apr/27/2017 | Adam Zess |  | Draft Release |

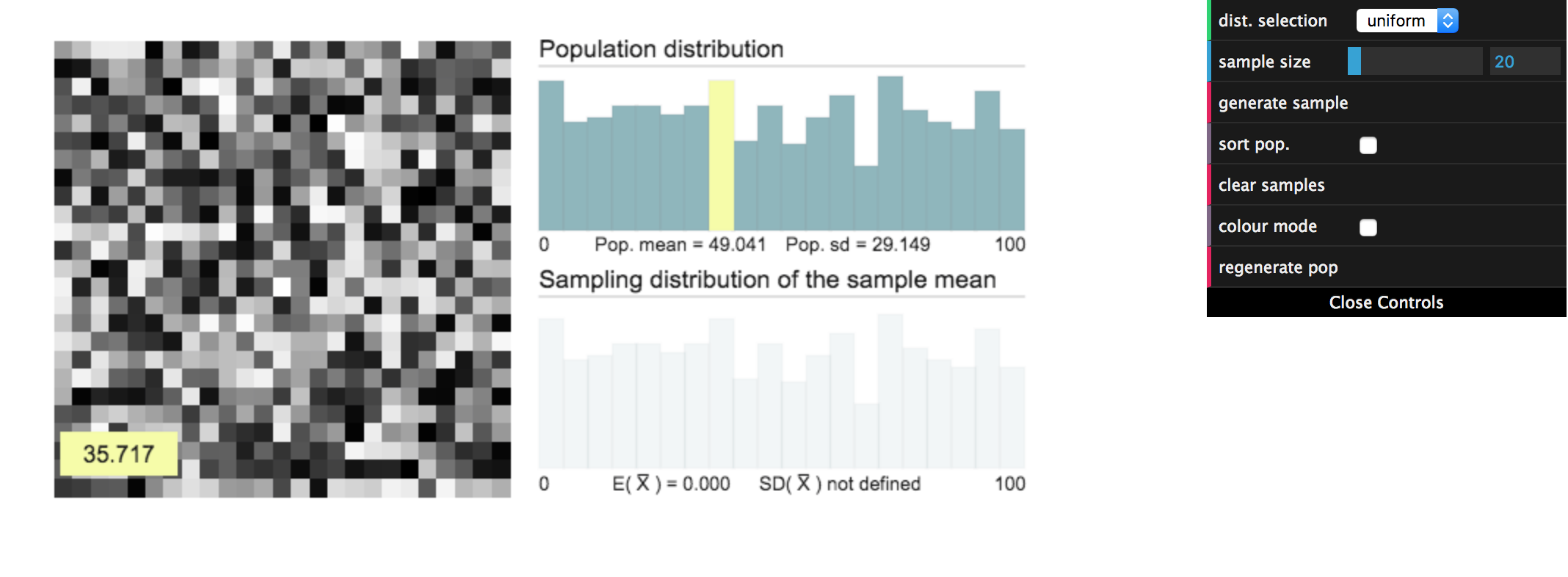
# **1. Introduction**

## ***1.1.* *Purpose and Scope***

The purpose of the Online Interactive Mathematical Visualization (OIMV) web-based course page project is to solve the issue of mathematical underpreparedness of students entering university studies - specifically mathematics courses and distinct concepts within these courses. This project aims to solve these issues through delivering mathematical lesson plans through dynamic online content with formative assessments and collaborative group-work. This document specifies the technical outline for the creation of the OIMV web site and gives a description of the various subsystems required for the device. Each subsystem and their respective technical requirements is explicitly explained in this document.

***1.2.* *Responsibility and Change Authority***

Each member is responsible for his own specifically assigned subsystem and for making sure that the requirements outlined in this document are met. Changes to the requirements towards this project will be made based off of agreement between each group members to avoid confusion and to remain transparent.



**Figure 1.** OIMV Example Visualization

**2. Applicable and Reference Documents**

## ***2.1.* *Applicable Documents***

The following documents, of the exact issue and revision shown, form a part of this specification to the extent specified herein:

|  |  |
| --- | --- |
| **Revision/Release Date** | **Document Title** |
| 8/3/2014 | Developing interactive mathematical visualisations |

## ***2.2.* *Reference Documents***

The following documents are reference documents utilized in the development of this specification. These documents do not form a part of this specification, and are not controlled by their reference herein.

|  |  |
| --- | --- |
| **Revision/Release Date** | **Document Title** |
| 8/3/2014 | Developing interactive mathematical visualisations |

***2.3.* *Order of Precedence***

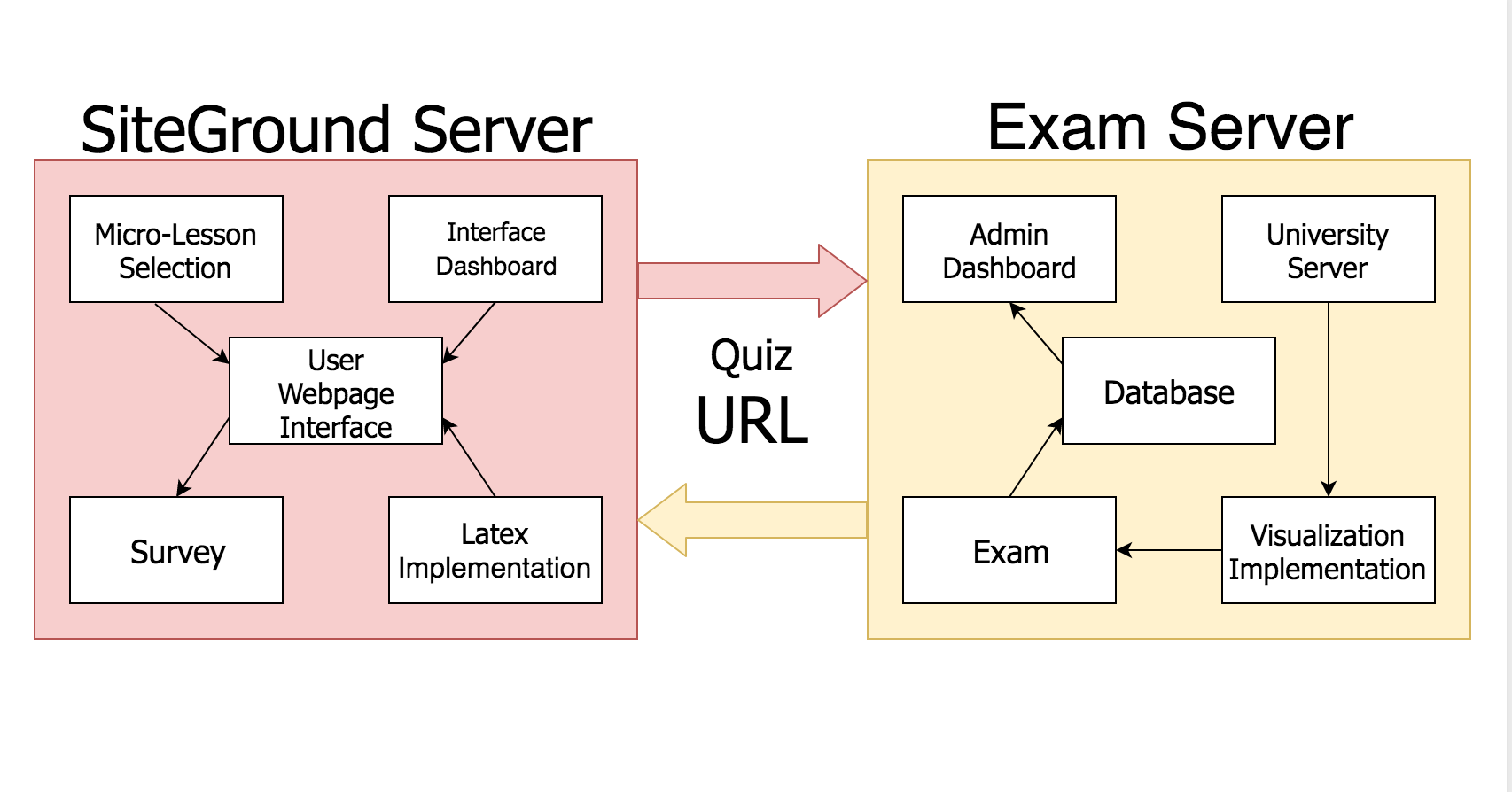
The OIMV website is to be used by students at Western Sydney University in Australia who are in entry-level mathematics course ‘Analysis of Change 1A,’ taught by Dr. Jim Pettigrew. The website will be used to teach specific micro-lessons that Dr. Pettigrew has deemed generally more challenging than the average concepts that are taught to students in his course. The students participate in the course online and take quizzes using the main OIMV to gauge the effectiveness of delivering contents to students in this way. The site should be used for educational purposes only and the results of the student’s efforts on the quizzes, exams and assignments will be sent directly to Dr. Jim Pettigrew, where he will be able to observe and analyzed the results.

**3. Requirements**

The Online Interactive Mathematical Visualization website will have two sets of functional requirements - requirements for the user (or student) and requirements for the site administrator (or professor). The user needs to have the ability to access the site, choose from different micro-lessons that are delivered by the site administrator, navigate throughout the micro-lesson to learn the content, participate in interactive check-your understanding prompts, pre-quizzes, post-quizzes, manipulate the OIMV to develop understanding of the mathematical concept being taught, complete a survey to gauge the effectiveness of the content delivery and have access to a review section to study previously delivered concepts. The site-admin must have the ability to observe the student’s results to the pre-test and post-test grades that each of his students made on the delivered assignments, view the amount of activity that each student committed to while completing the module, as well as have the ability to manipulate the student-user interface by adding new content simply and intuitively.

## ***3.1.* *System Definition***

Our OIMV web system will consist of two distinct subsystems - the student user interface subsystem and the administrator interface subsystem. The student user interface subsystem includes the website hosted on the SiteGround server on the WordPress content management system and includes the webpage that students will be able to navigate through directly to follow the structure of the course and deliver the learning content.The administrator interface subsystem hosts the examinations as well as the embedded mathematical visualization that will be recorded and sent to the administrator portal database. The admin subsystem will also include a dashboard in which the site admin can observe the results of the students efforts on the quizzes, as well as the activity that each student performed with the visualizations. The subsystems of the OIMV system are shown in Figure 2 and include the student user interface subsystem (hosted on the SiteGround server) and the administrator interface (hosted on the exam server).

****

**Figure 2:** Online Interactive Mathematical Visualizations System Block Diagram

***3.2.* *Characteristics***

### **3.2.1. Functional / Performance Requirements**

#### **3.2.1.1. Student Web Navigation**

Students are able to access the course page and navigate through the different micro-lesson submodules, complete the required portions after learning the content, and review the material that was delivered.

#### **3.2.1.2. Exam Module Access**

Students should have access to different types of examinations including check-your- understanding prompts, pre quizzes and post quizzes. The students should be able to login so their results are recorded and submitted to the site admin, as well as answer the questions and view their grade on the sections with feedback.

#### **3.2.1.3. Survey Responses and Review**

Students should have the ability to complete a customizable survey delivered by the site administrator to gauge the student’s responsiveness to the course material deliverance. The students should also have access to the delivered content in the form of a review tab that can be accessed and revisited at a later time by the students.

#### **3.2.1.4. Administrator Dashboard**

The administrator should have access to a dashboard that produces tabulated results of the students efforts on the examinations as well as the amount of activity each student performed on the visualizations, which corresponds to the time spent manipulating the visualization.

#### **3.2.1.5. Administrator Content Implementation**

The administrator of the site should have the ability to implement future content to the student user interface intuitively when future content is developed. He should be able to add future micro-lessons easily and add functionality to the exam recording implementation of the site as well.

#### **3.2.2. Outputs**

##### **3.2.2.1 Student Interface Website Structure**

Students have the ability to successfully navigate through the sequential models of the student interface webpage in order to learn the content and complete the required modules specified by the site administrator for each micro-lesson.

##### **3.2.2.2 Examination Results**

The student’s results produced from their attempts on the quizzes should be recorded and sent to the administrator database to be viewed on the admin dashboard in tabulated form for analysis by the site administrator and input into the university grade database.

##### **3.2.2.3 Student Visualization Activity**

The activity performed by each student on the visualization should be recorded and stored as the number of mouse activities that each student performs with the online interactive mathematical visualization that is embedded within the post quiz to help the students with an alternative method of learning.

# **4. Support Requirements**

Our group has decided that we shall create an admin manual to help admins with the back-end files upload as well as the interface that the admins will have to use to manage the database correctly. The admin manual will provide detailed instructions of the visual interface display of the admin portal and provide descriptions for each function and of the operations that the admin can perform with the website. This manual will allow admins to have a complete understanding of the website and if any question arises, they will be assured to have an answer that they can access themselves. There will also be a video instruction provided for the website using, including component listings and data examples regarding the different functions in the website. This is beneficial in replacing parts and lists the specific technical properties and the requirements that replacement parts must adhere to.

# 

# 

# 

# **Appendix A Acronyms and Abbreviations**

OIMV Online Interactive Mathematical Visualization

Online Interactive Mathematical Visualizations

Adam Zess

Sen Yang

**Interface Control Document**

REVISION – Draft

27 April 2018

Interface Control Document

for

OIMV

Prepared by:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Author Date

Approved by:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Project Leader Date

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Prof. Stavros Kalafatis Date

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

T/A Date

**Change Record**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Rev.** | **Date** | **Originator** | **Approvals** | **Description** |
| **-** | 4/27/2018 | Sen Yang |  | Draft Release |

# **Overview**

This document, ICD (Interface Control Document), describes the interactive interface of the OIMV (Online Interactive Mathematical Visualization) website. This project is set up at the beginning of the semester, our team will create and add functions to a new website within one semester. The two subsystems are the front-end subsystem and back-end subsystem. The two team members will take charge of each subsystem. Taking advantage of the reference materials that might give insight to further our project design, our team set up the outline of the interface for each different field. Considering the pros and cons of different options, our team deliberately decided what the details of each interface will be. This website design is a pure software project, we don’t have hardware components aside from the use of the website by students on their personal devices. The user interface will consist of micro-lessons, lesson submodules, exams, and surveys. To collect data remotely, interactive interface protocols will be needed. The ICD will describe host service, user interface, and admin interface.

# **References and Definitions**

Our sole reference, delivered to us through the efforts of our sponsor and future web-administrator, has been used to obtain background information regarding the OIMV and its theoretical uses. The student user interface subsystem utilizes a SiteGround server to host the WordPress content management web-page that students have the ability to navigate through to learn the material. The student interface subsystem is connected to the admin back-end subsystem through the quiz that is delivered on the front end portion of the website, in which the results are obtained and stored on the back end of the website. The admin back-end subsystem uses part of the ‘Super Exam System’ based on Django and the exams are created by Dr. Jim Pettigrew, who is our sponsor and has provided specific contents for his teachings. The mathematical visualizations are created by Canvas and are embedded within the quiz modules.

## ***References***

[1] J. Pettigrew *et al.*, “Developing interactive mathematical visualisations” Mathematics Education Support Hub., University of Western Sydney, Sydney, Australia, 3, August 2014.

## ***Definitions***

OIMV Online Interactive Mathematical Visualization

## 

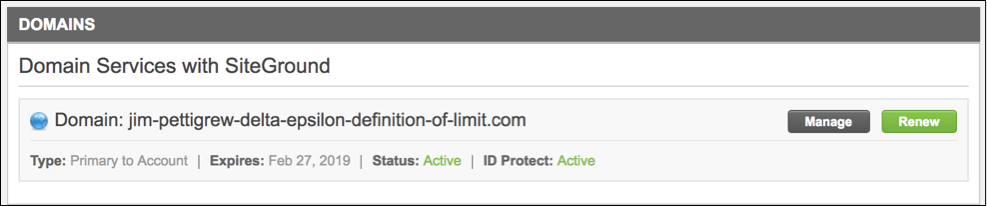
# **Software Interface**

## ***Student-User Interface***

The student-user interface is established to hold the site that will be accessible to students for the deliverance of the content as well as provide students access to to the check-your- understanding prompts, pre quizzes, post quizzes, and survey responses.

### **SiteGround Web Service**

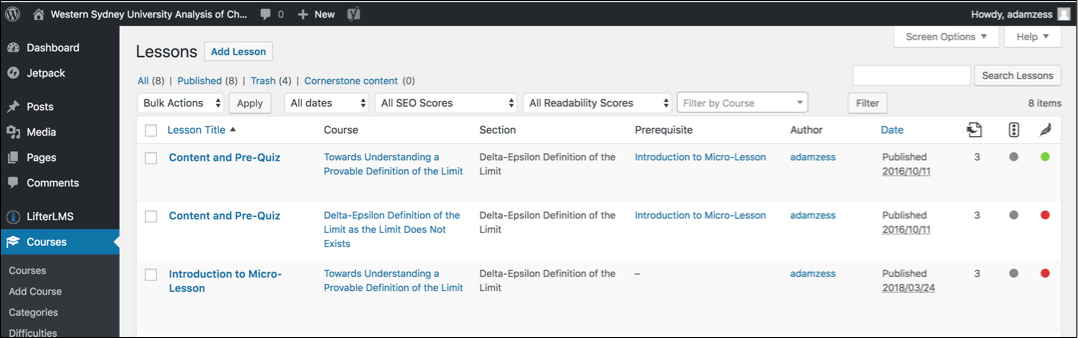
The student-user interface is hosted on a SiteGround server. The advantages to hosting the server with this service with this service are that the geographical proximity of the server to the users leads to low latency when used. The server is also reliable, safe, and relatively inexpensive and very intuitive to manage.



**Figure 1:** SiteGround Administrator Control Panel

### **WordPress Content Management System**

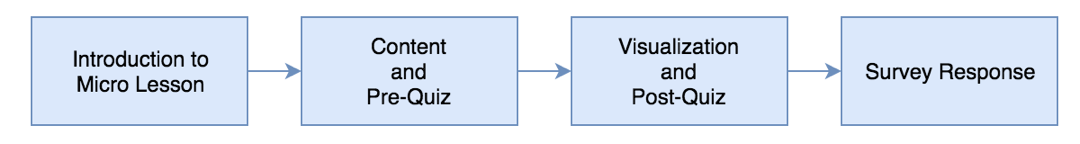
The student-interface is managed using the WordPress content management system. WordPress is very compatible with the SiteGround server. WordPress is also intuitive to learn and our sponsor will be able to revise and create additional content for the site based on the existing structure of the course site.



**Figure 2:** WordPress Content Management System Admin Dashboard

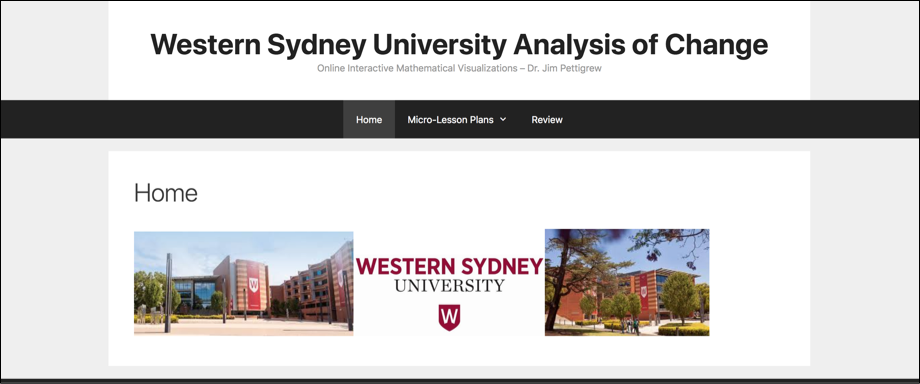
* + 1. **Website Conceptual Flow and Overview**

Each micro-lesson delivered to the student has different, sequential modules for the introduction, content and pre-quiz, visualization and post quiz, and survey responses.



**Figure 4:** Student Interface Micro-Lesson Conceptual Flowchart

The student-user interface of the course page can be shown below. The student has access to the homepage, micro-lesson selection and review page. When a micro-lesson is accessed, the student has the ability to navigate through the micro-lesson, review the content, manipulate the interactive visualization take the exams, and complete the survey.



**Figure 4:** Student Interface Web-Page Display

## ***Administrator and Exam Interface***

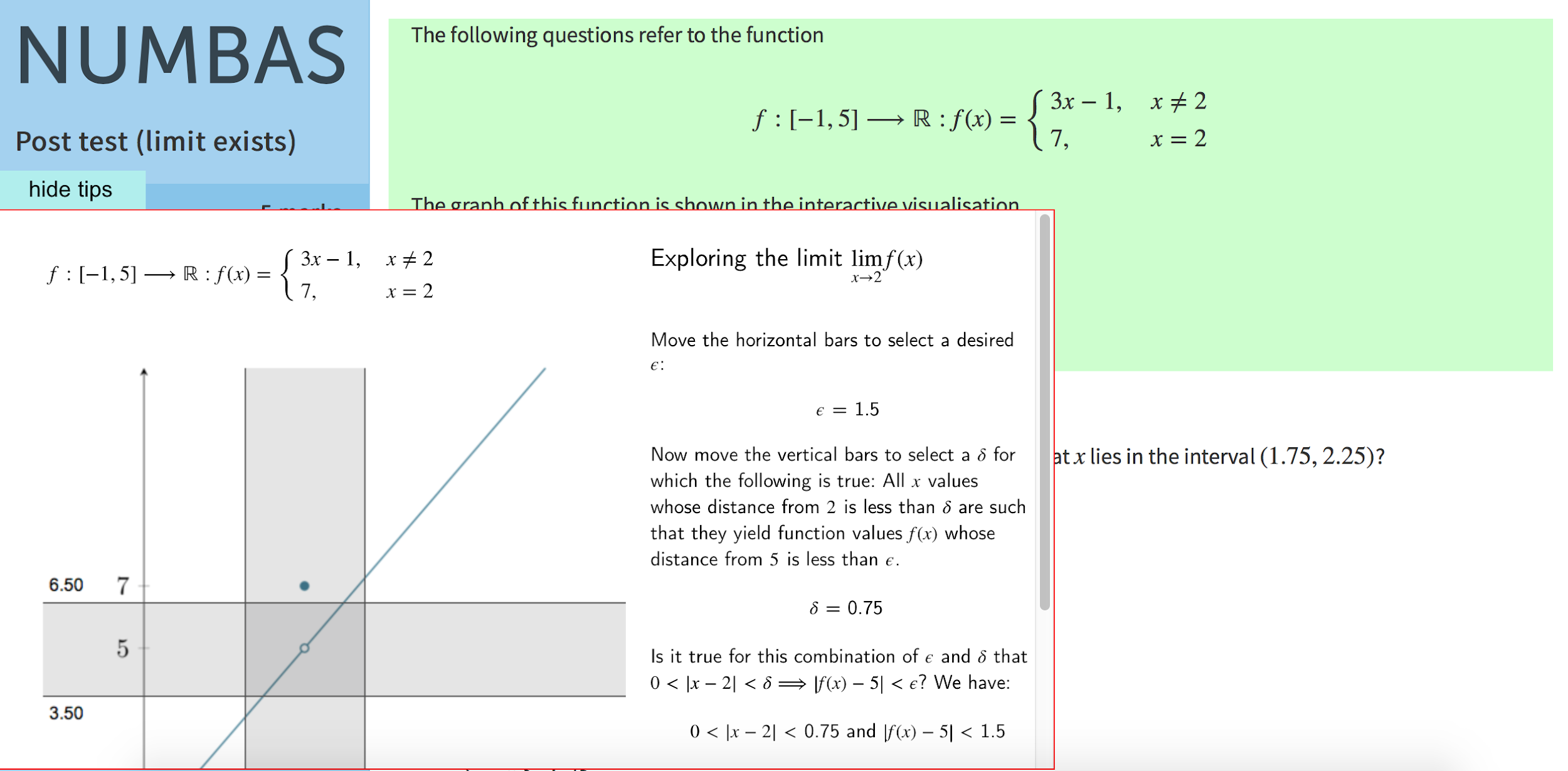
The student-user interface is hosted on a SiteGround server. The advantages to hosting the server with this service with this service are that the geographical proximity of the server to the users leads to low latency when used. The server is also reliable, safe, and relatively inexpensive and very intuitive to manage.

### **Exam Server**

The exam subsystem uses Ajax, Python, and MySQL framework to achieve proper functionality. It uses Python to receive the the exam results and information from the front-end of the site and saves the results to the exam database.  
  
The front-end uses iframe.js and ajax.js to save the user’s mouse activity with the visualization and send it to the exam database.  
  
When a user logs into the exam using the URL in WordPress, the browser will track the user information. If the user information is not existing, it will ask the user to enter their login information; if the user information is already existing, it will directly jump to the homepage of the exam system, and the student takes the exam.

### **Exam and Visualization Implementation**

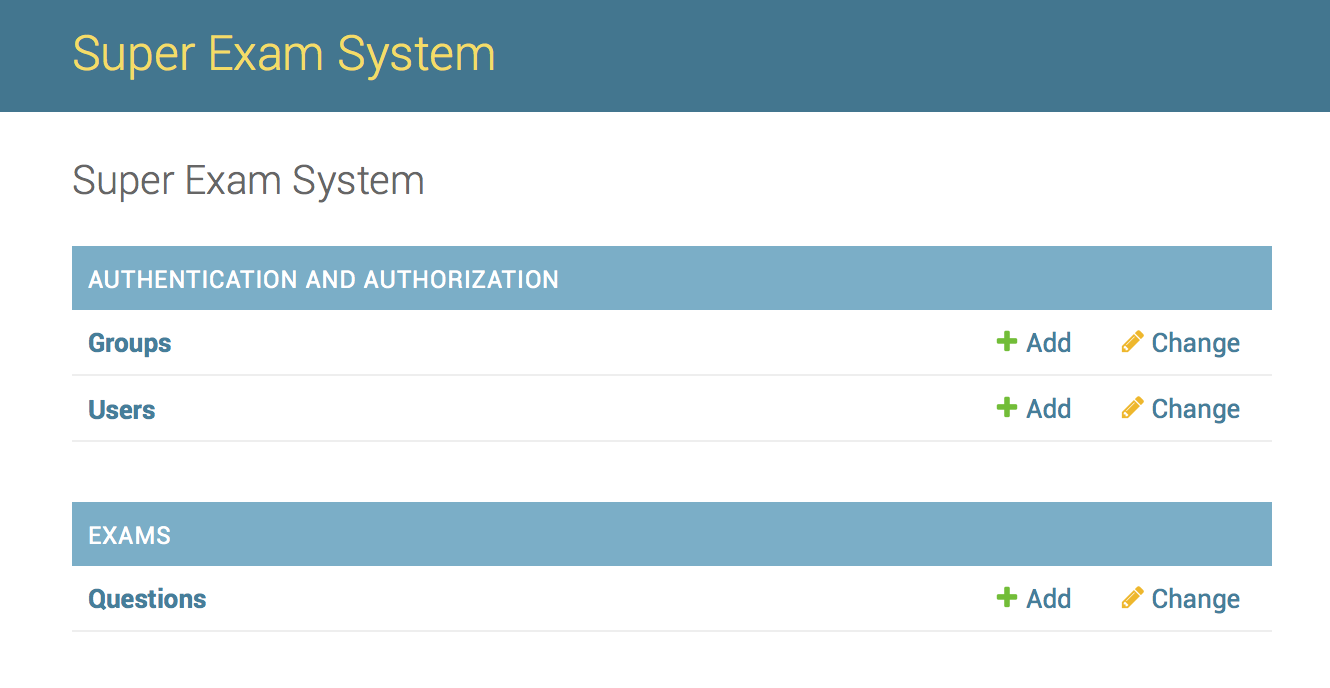
The student can access the exam through the student-user interface. The user then logs in so their responses can be recorded and sent to the database. The exam is entered and the user has the ability to manipulate and utilize the visualization’s features to answer the exam questions.



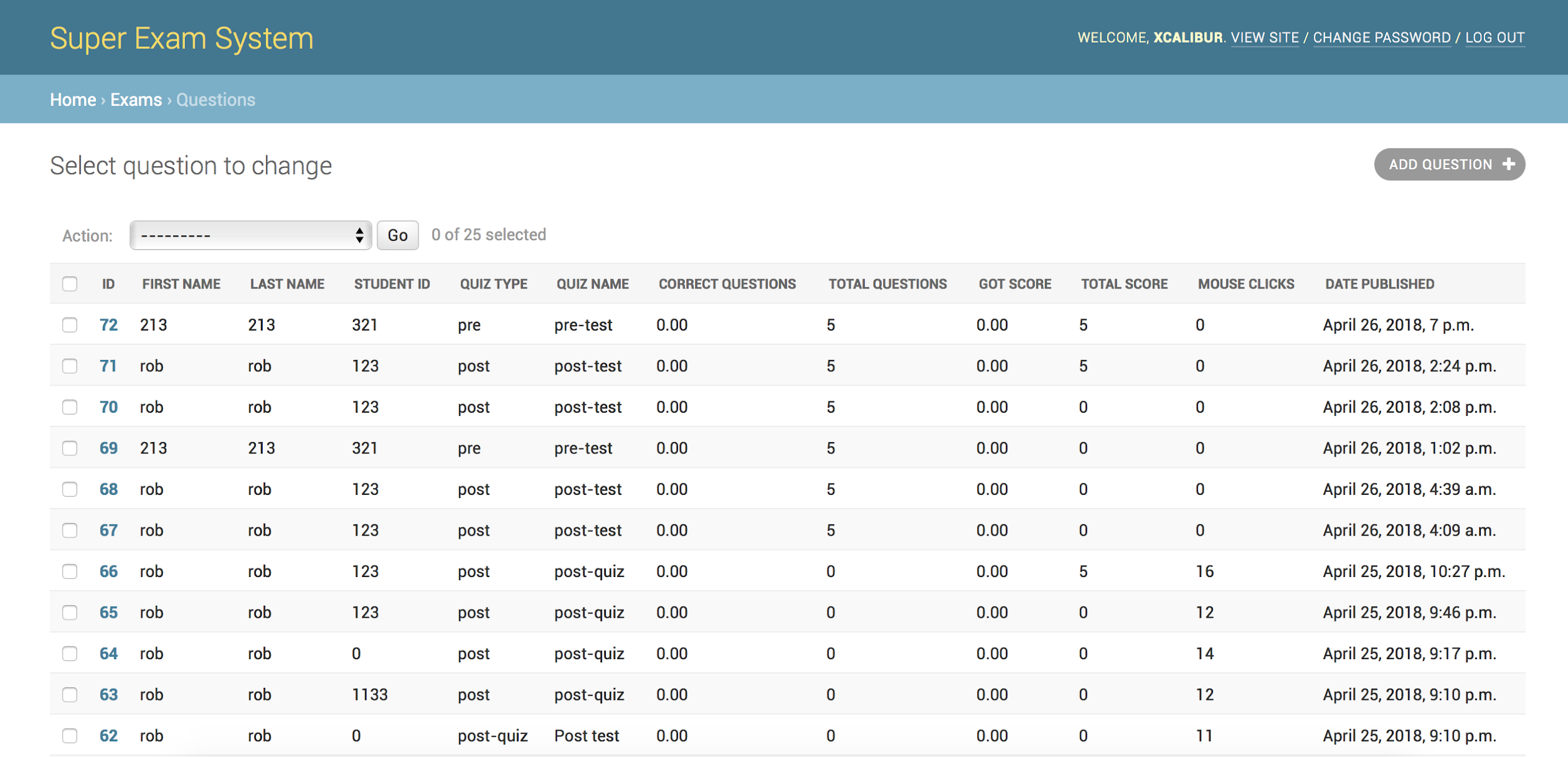
**Figure 3:** Examination and Visualization Display

* + 1. **Administrator Database Portal and Dashboard**

The website owner can enter the admin-portal through the login-page. From the portal, the admin has the ability to manage the database by adding other site admins and observing and editing the student’s results from the quizzes.



**Figure 4:** Administrator Control Panel



**Figure 4:** Administrator Database Panel

**APPENDIX C – SUBSYSTEM REPORTS**

Online Interactive Mathematical Visualization

Student-User Interface Subsystem

Adam Zess

**Final Report**

1. **Subsystem Introduction**

The front-end student-interface subsystem of the Online Interactive Mathematical Visualization project is the subsystem of the project that hosts the portion of the site that students will have direct access too. The goal of this subsystem was to create a web environment that students could access that provided students the ability to navigate an aesthetically-pleasing interface with seamless integration of mathematical notation rendering. The students should have access to a menu that allows them to access a home tab, micro-lesson access submenu, and review tab to review the information and quizzes delivered to them within the course. When a micro-lesson is accessed, the student has the ability to navigate through the micro-lesson, review the content, manipulate the interactive visualization take the exams, and complete the survey. Another important feature for the front-end of the site is reliable hosting ability and intuitive content management system allows for intuitive manipulation and revisability by the future site administrator for the site.

**2. Subsystem Overview**

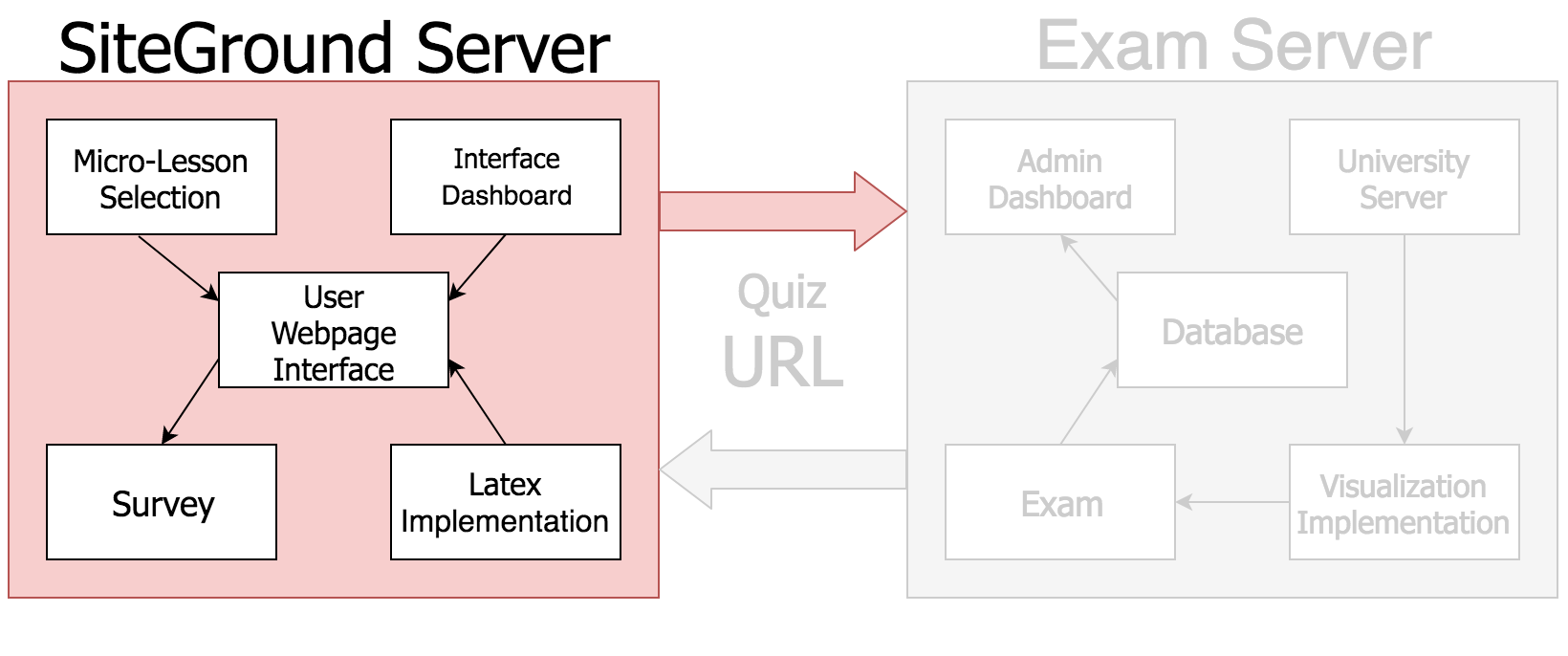
****

Figure 1: Student-User Interface Subsystem Block Diagram

1. SiteGround Web Hosting

The Student-User Interface Subsystem is hosted on a SiteGround Server. There are multiple reasons why using this hosting service for the Online Interactive Mathematical Visualization site was optimal. The physical location of the server farm allows for reduced latency and higher performance of the site when students use and operate the site. SiteGround is also very reliable web hosting service with very positive user feedback and 24/7 customer support so the future site administrator will be at ease with the knowledge of using a high-performing web hosting service. For the benefits that the web hosting service provides, the server is also relatively inexpensive compared to their main competitors and overall provided the best options for hosting this project in the future.

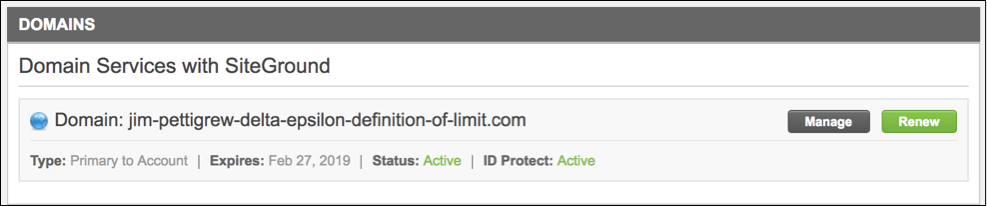


Figure 2: SiteGround Web Hosting Administrator Control Panel

1. WordPress Content Management System

The WordPress content management system provided the Online Interactive Mathematical Visualizations student-user interface subsystem with the best option towards delivering an aesthetically-pleasing user display that can be designed flexibly with interchangeable parts. The WordPress content management system works very well with the pre-established SiteGround web hosting and is another ease-of-use establishment to provide the future site administrator with an intuitive approach to revise existing content within the site and add additional content to the site when developed. WordPress comes with an administrator dashboard that holds the front-end development of the site together and has additional plug-in options for future modifications of the site based on the preferences of our sponsor.

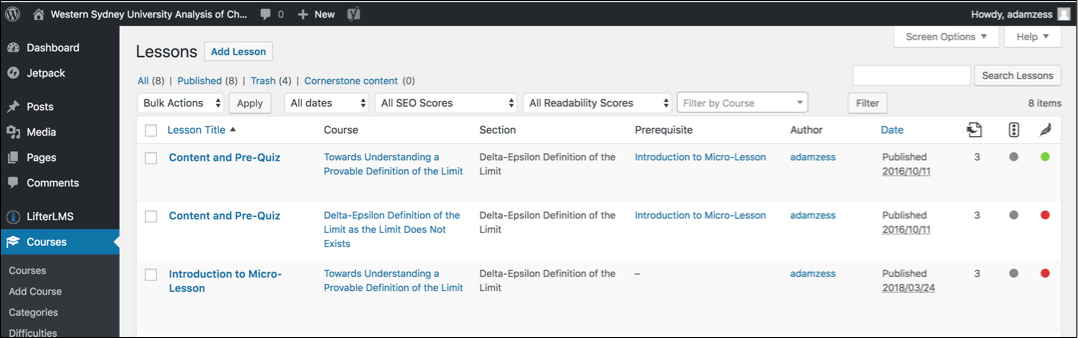


Figure 2: WordPress Administrator Featured Dashboard

**3. Data Collected**

The front-end development and visual displays that are portrayed to the students that the site is designed for are the primary collected data points for the student-user interface subsystem. The student-user interface was designed in a way that gave students the ability to enter the page at the home tab of the website and immediately be granted access to the customizable main menu. Working with our sponsor to conceptual plot a plan for the delivery of the main front-end components of the website, the main menu currently has a home tab, a micro-lesson selection mini-menu and a review tab. The review tab gives students the option to review the collection of check-your- understanding prompts, pre quizzes, post quizzes, as well as the visualization to give students the ability to review content learned within the course. The micro-lesson selection mini-menu is the most important portion of the student-user interface. This mini-menu allows student to select from the different developed micro-lessons that have been developed and integrated within the course website. These menus as well as the structure of the website were designed and can be manipulated within the WordPress control panel.

When a micro-lesson is accessed, the student has the ability to navigate through the micro-lesson, review the content, manipulate the interactive visualization take the exams, and complete the survey.

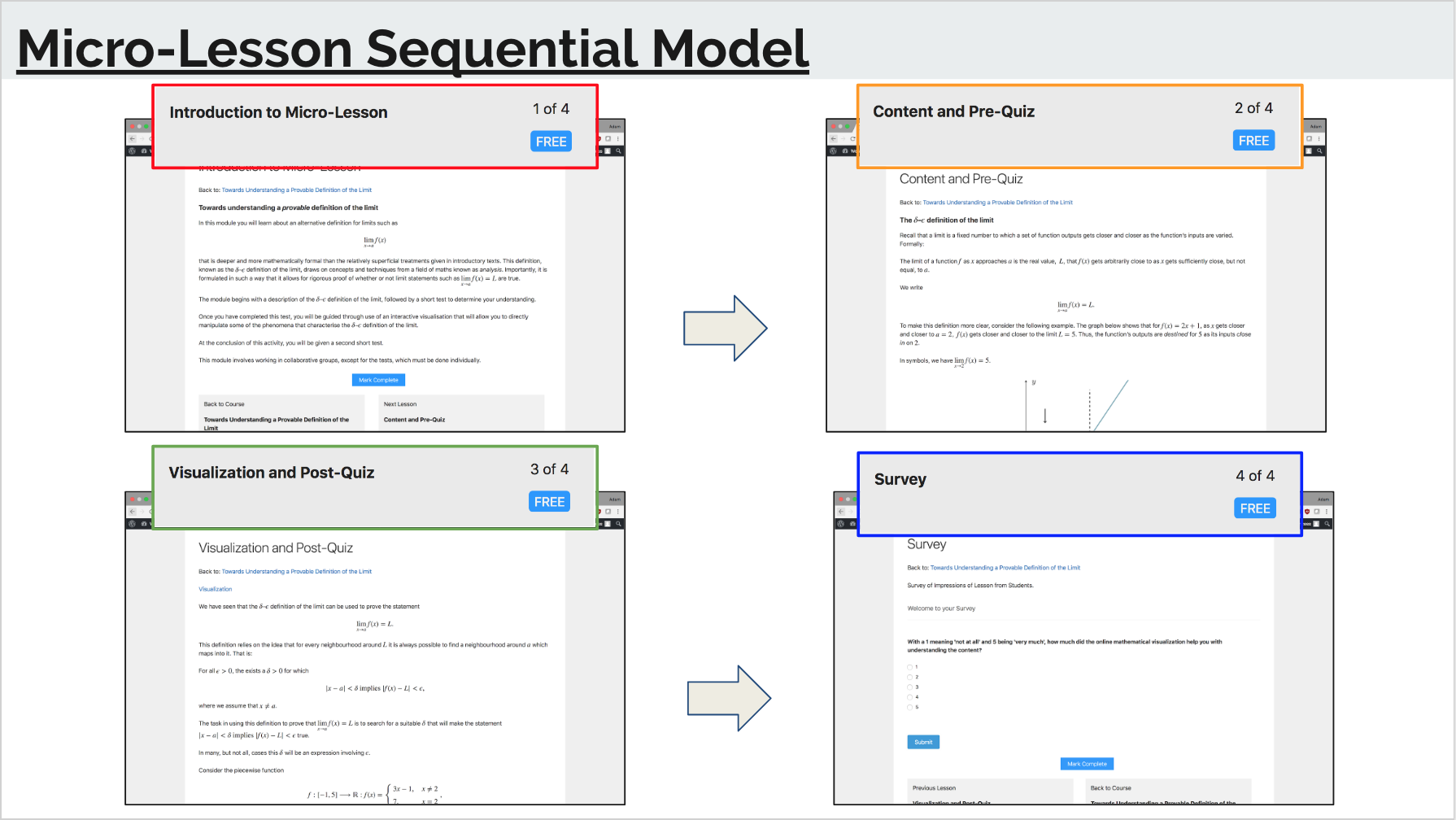


Figure 3: Sequential Micro-Lesson Module Content

**4. Validation Plan**

The validation plan prescribed for testing the successful implementation of this subsystem will be more focused on the deliverable, displayable results that are produced from the structural design of the website and the delivery and integration of content within the website. The focus of this subsystem’s deliverables are the seamless integration of mathematical notation rendering as well as the proper sequential flow of the website that students use to navigate through the site and each specific micro-lesson. The intuitive nature of the WordPress content management system allows for the inherent operation of the site components through the front-end and back-end integration with the use of the content management system, so this eliminates one potential problem that could have arisen without use of this system.

The seamless integration of mathematical notation rendering was done through the use of MathJax/Latex implementation throughout the framework of the body of the site. Each page that is designed using Latex shortcode delimiters is designed to produce specified typography displays corresponding to different mathematical equations. This process can be shown below using the Latex shortcode - a mathematical notation rendering language that is used to transcribe short-coded mathematical formulas into beautifully-rendered mathematical equations. The proper functionality of this subsystem requirement can be demonstrated below:

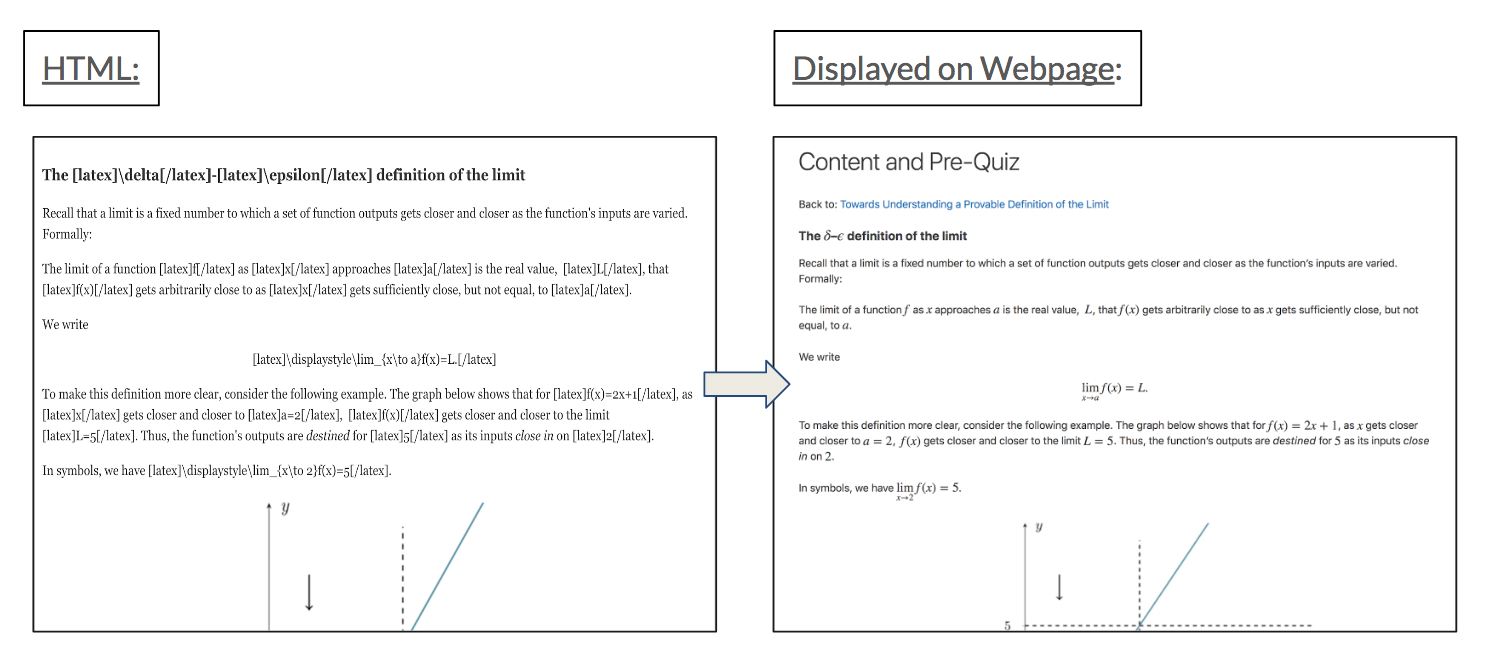


Figure. 5 User Information Collection Test

The conversion of the short-code written in the HTML/CSS portion of the WordPress administrator dashboard can be shown to produce transcribed mathematical equations that are seamlessly integrated into the content-delivery portion of the micro-lessons within the student-user interface.

Online Interactive Mathematical Visualization

Administrator Interface Subsystem

Sen Yang

**Final Report**

1. **Subsystem Introduction**

The Online Interactive Mathematical Visualization project aims to address the problem of the mathematical under-preparedness of students transitioning into university study, and the risk this poses to their engagement. It seeks to improve students’ engagement with their mathematical studies by allowing them to collaboratively explore visual representations of important concepts in the discipline. A further aim is to pilot the use of these visualization tools as examples of an approach to curriculum that combines dynamic online content with opportunities for socially constructed learning and formative assessment.

The back-end subsystem of the Online Interactive Mathematical Visualization project is made up of an exam system, an admin management system, a user login system, as well as a visualization system. The goal of this subsystem is to record students’ information, exam data, and interactive activities when they take the exams. And based on these records, the professor can summarize if the mathematical visualization teaching is helpful for students’ mathematics study.

**2. Subsystem Overview**

The most significant part of the back-end subsystem is the exam server. An Asia server was selected as it was approaching to the Australia. Since the WordPress is written by PHP while the exam was created by NUMBAS written by HTML, CSS, and JavaScript, the exam cannot be embedded directly to the WordPress. The exam server holds all the back-end stuff, which includes the exam system, the admin management system, as well as server configuration system.

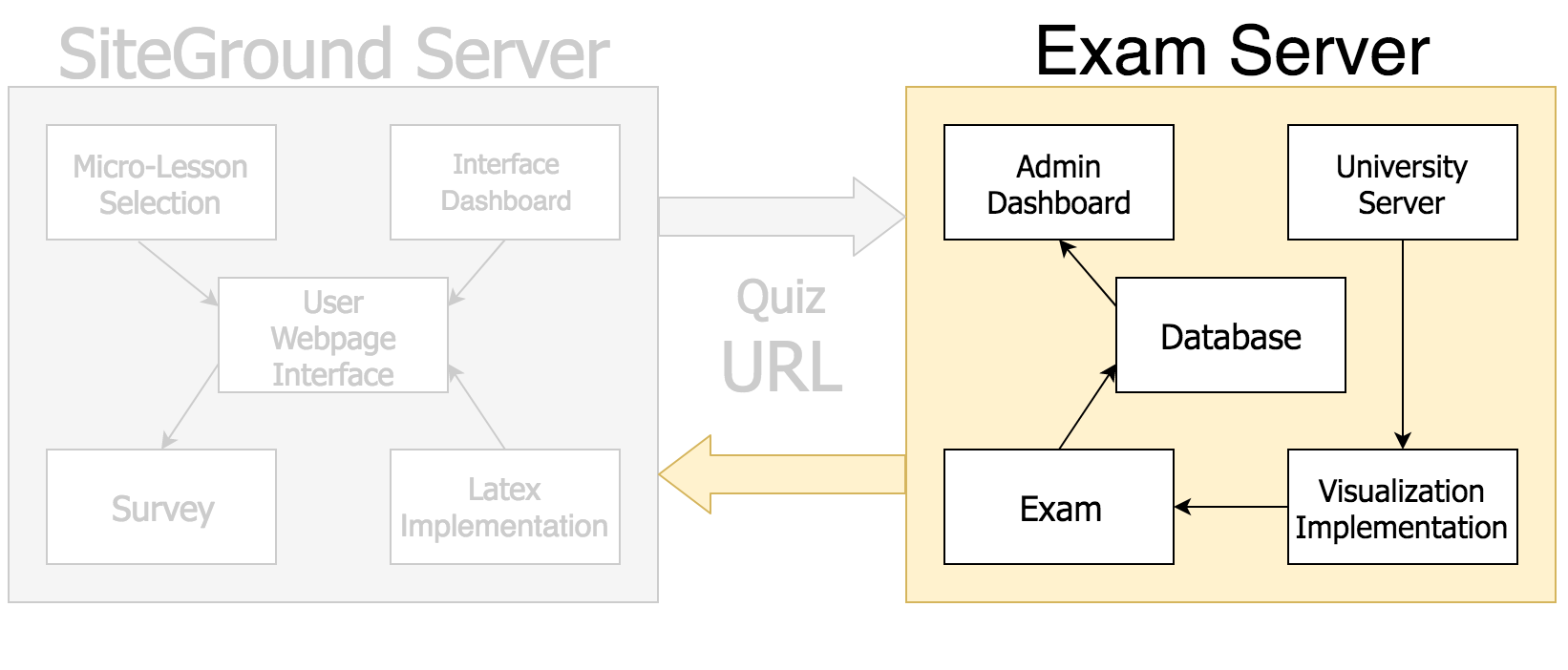


Figure 1: Administrator Interface Subsystem Block Diagram

**3. Data Collected**

When a user logs into the exam using the URL in WordPress, the browser will track the user information. If the user information is not existing, it will ask the user to enter their login information; if the user information is already existing, it will directly jump to the homepage of the exam system, and the student takes the exam. When the students enter the exam, the iframe.js can record their grade report as well as the interactive mouse activities, then the Ajax will send these data to the back-end system as save them in the database.

Figure 2: User Login and Delivery of Post Quiz and Visualization

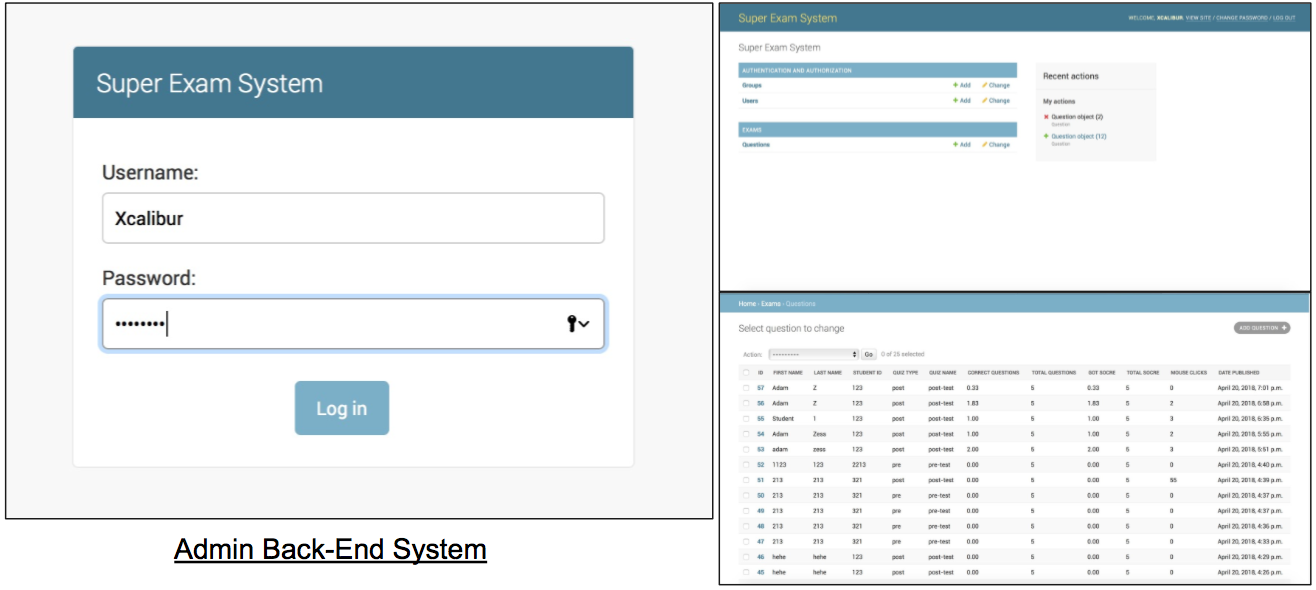


Figure 3: Administrator Login for Managing Exam Data

**4. Validation Plan**

The validation plan for this subsystem is to test if the user information and their activities can be recorded and saved in the database. The values shown in Figure 5 is to test if the user port open for recording user information while the values shown in Figure 6 is to test if the database port open for recording user mouse activities in the database.

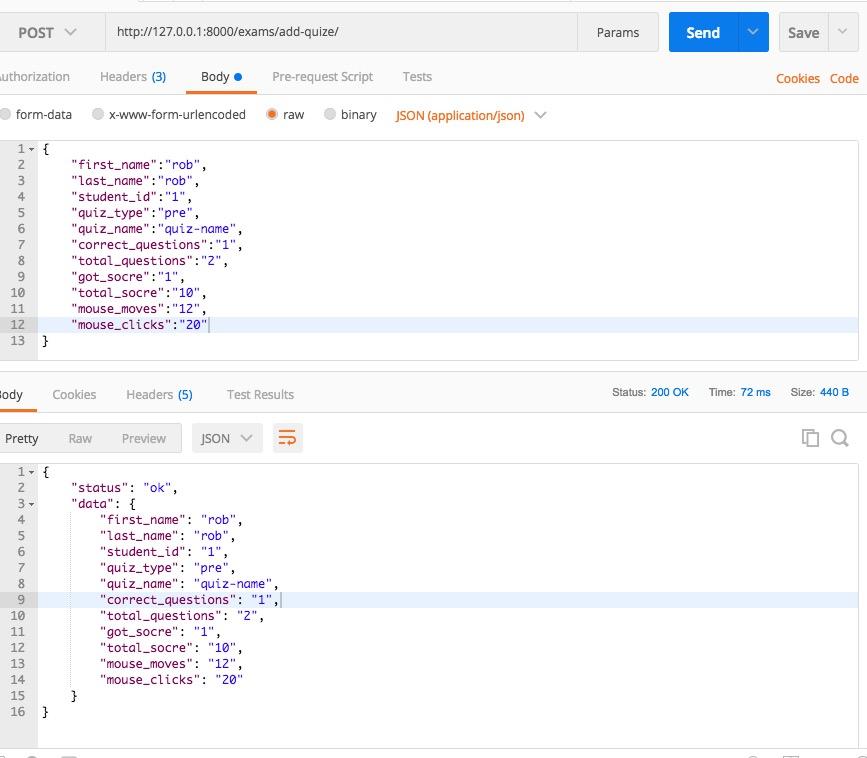


Figure 4: User Information Collection Test

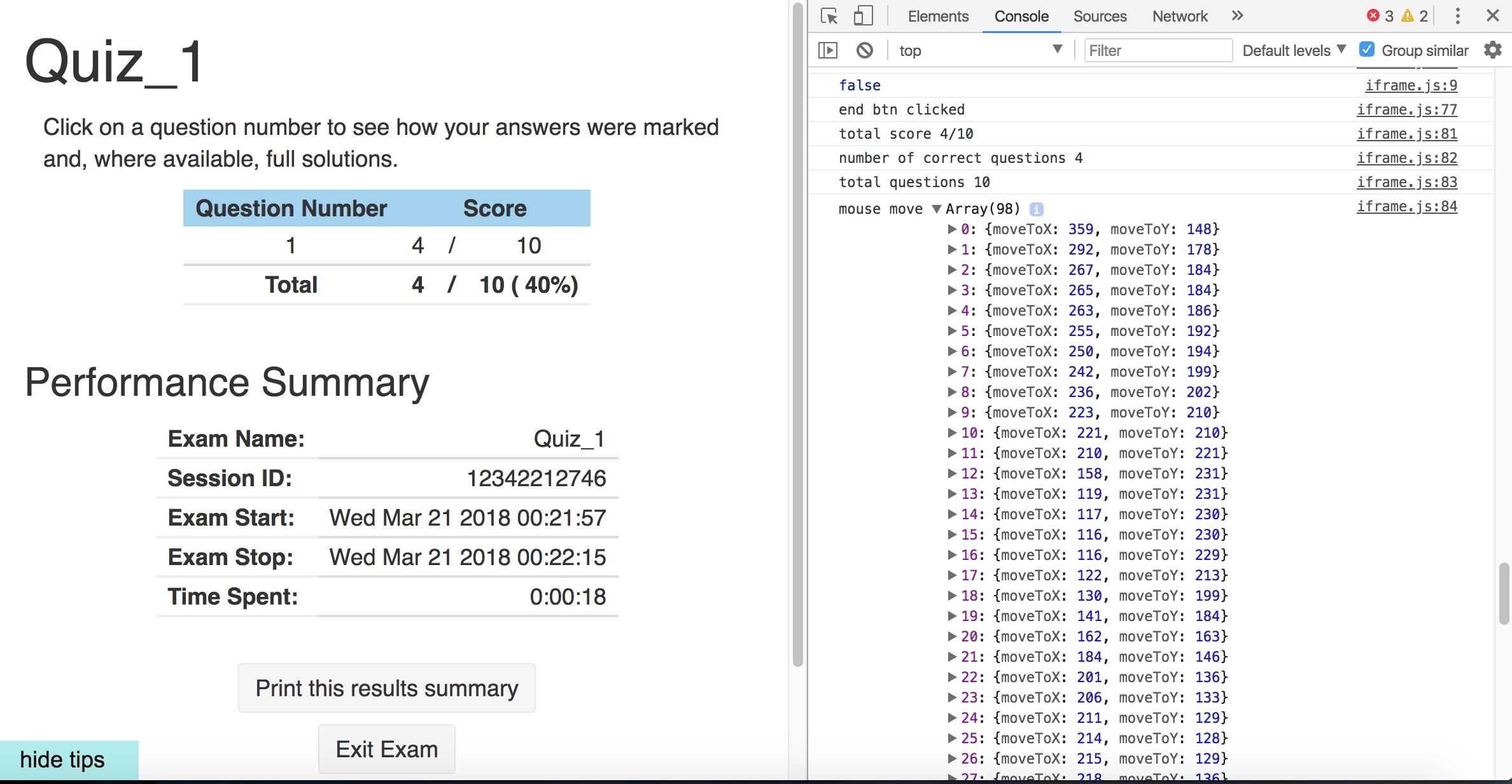


Figure 5: User Interactive Mouse Activities

Online Interactive Mathematical Visualization

Adam Zess

Sen Yang

**SYSTEM DESCRIPTION AND DEVELOPMENT**

**(GROUP REPORT)**

REVISION – 29 April 2018

1. **Overview**

This project proposal aims at describing the design, test, and validation of a website that has the capability to deliver dynamic online content to students to provide an alternative method of teaching difficult mathematical material. The primary goal of the Online Interactive Mathematical Visualization (OIMV) web system is to combat mathematical underpreparedness of students entering university studies and to improve participation of students learning mathematical concepts and development of mathematical understanding and learning habits. There are many online course systems that currently exist on the market, however, this system is different because the project aims to deliver an interactive mathematical visualization that students can work on collaboratively to gain a more firm understanding of the mathematical concepts being delivered to them. The micro-lesson material will be delivered to the students via a student-user interface site, that students can access to navigate through different micro-lessons and learn content, take part in quizzes, access visualizations and review material. The site administrator has access to the results of the students efforts in the form of a admin portal dashboard that displays the grades that students received on their examinations and their corresponding interaction activity level with the visualization. The administrator will also have the ability to develop both front-end and back-end content intuitively on the system and can add new content and revise existing content on the website for both the student-user interface and the admin-interface.

The Online Interactive Mathematical Visualization web-based project is being developed for Dr. Jim Pettigrew of the University of Western Sydney to be used to implement and teach specific micro-lessons for content that has been deemed unintuitive and requires further educational instruction than just traditional classroom learning. The website is centered along the use of the online interactive mathematical visualization that allows students to access and the micro-lesson webpage, be delivered content and take a pre-quiz to test their understanding of the material following traditional classroom delivery of the content. Following the submission of the prequiz, students then are provided access to the visualization and postquiz and the results of their efforts on the postquiz are examined to analyze the effectiveness of delivering content via the interactive visualization modules.

1. **Development Plan and Execution**

**2.1 Plan**

This project aims to address the problem of the mathematical under-preparedness of students transitioning into university study, and the risk this poses to their engagement. It seeks to improve students’ engagement with their mathematical studies by allowing them to collaboratively explore visual representations of important concepts in the discipline. A further aim is to pilot the use of these visualisation tools as examples of an approach to curriculum that combines dynamic online content with opportunities for socially constructed learning and formative assessment.

There are already online course delivery systems that exist for many different types of courses online for many different scopes and classes. The main difference between existing systems and this system is the introduction of dynamic content through the implementation of the interactive mathematical visualization. The site if primarily being developed for WSU students studying mathematics units such as Analysis of Change, Mathematics 1A, or any of a range of first level statistics units. Depending on the project’s generalisability, potential users outside of the mathematics and statistics disciplines areas could be engaged, both at WSU and other universities. The visualization-centric web page will be used as an experiment to test students understanding of material after deliverance of the dynamic online interactive visualization.

Dr. Jim Pettigrew of the University of Western Sydney has hypothesized that students have trouble when entering university study, specifically in math courses with individual micro-lessons that have not been effectively taught to students. This site will be implemented to deliver content in a dynamic method to test the effectiveness of this teaching style on students understanding of the material.

**2.2 Data**

*A. Proposed System Diagrams*

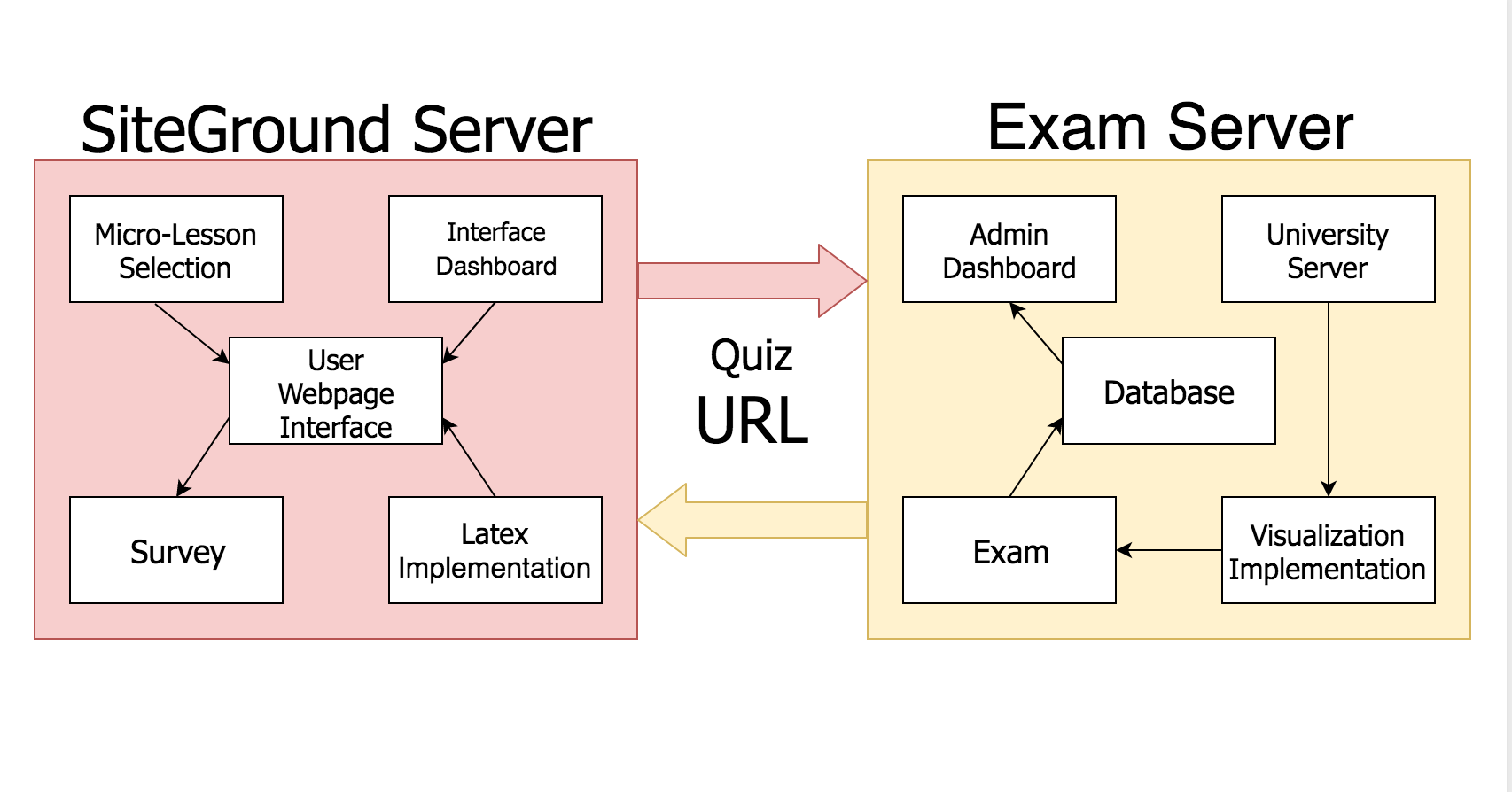
**

Figure 1: Online Interactive Mathematical Visualization System Block Diagram

*B. Project Deliverables*

1. System Functions or Features: The website functions specifically in two parts - a student-user interface where students have access to the course content being delivered and the administrator-interface where the administrator has access to a database in the form of a dashboard that shows each students results from using the quizzes and visualizatoins.
2. Project Deliverables: An aesthetically pleasing student-interface webpage that has the ability to host different micro-lesson plans and can be navigable sequentially with check- your-understanding prompts, pre quizzes, post quizzes with visualization, survey responses and a review tab. The student-interface needs to deliver content sequentially with seamless mathematical notation rendering on the site. The administrator needs to have an interface that displays the results of the quizzes obtained by the students with their corresponding usage of the visualization. The administrator needs to also have the ability to manipulate both the front-end and back-end development of the system intuitively.
3. Preliminary Test Plan:
   1. Demonstrate the navigation of the front-end development of the student interface.
   2. Demonstrate the delivery of the back-end development of the administrator portal.
4. Final Test Plan: Combination of the the two subsystems leads to a fully functional product that the site administrator can use to deliver and measure results of students uses of the site with online interactive mathematical visualizations.

C. ABET Constraint Considerations

1. Health and Safety: The website is an entirely web-hosted and software system so the system is not responsible for any health and safety considerations.
2. Environmental: The website is an entirely web-hosted and software system so the system is not responsible for any environmental considerations.
3. Economical: The functional costs of the website consist of the server cost and domain purchase for the site. The SiteGround web hosting service is relatively and cheap and inexpensive based on the reliability that the service provides. The Exam Server is cheap and is designed and hosted for back-end development with extreme flexibility.
4. Ethical: The website development is ethical and used to further students understanding of complex or unintuitive concepts, especially in mathematics. The main aim of the project is the improvement of students understanding of concepts and access to the site does not include and bonus costs to the students.
5. Manufacturability: The website manufacturability is repeatable and relatively intuitive. The site administrator will have the ability to host this site on future domains for different courses and have the ability to format the site to specific preferences, revise existing micro-lessons and content, as well as add additional micro-lessons and content to the site when specific content is developed, based off templates for front-end and back-end development.

**2.3 Conclusion**

In conclusion, the goals of our web project were achieved through the delivery of the online interactive mathematical visualization site. The site is fully functional and can be successfully operated to deliver dynamic content to the users and record the results of their usage of the exam and visualization to deliver to the site administrator. The website admin has the ability to add future courses to the site and the site allows him to develop the front-end and back-end functionality of future courses simply and intuitively using the existing methods used in this project.

**Appendix D**

##### **Subsystem <Administrator Interface> Code**

#Admin.py

from django.contrib import admin

from .models import Question

class QuestionAdmin(admin.ModelAdmin):

# list\_display = ('first\_name', 'last\_name','student\_id','quiz\_type','quiz\_name','correct\_number','total\_number','get\_socre',

# 'total\_socre','mouse\_move','mouse\_click')

list\_display = [field.name for field in Question.\_meta.get\_fields()]

list\_per\_page = 25

admin.site.register(Question,QuestionAdmin)

#models.py

from django.db import models

class Question(models.Model):

first\_name = models.CharField(max\_length=20,default='rob')

last\_name = models.CharField(max\_length=20,default='rob')

student\_id = models.PositiveSmallIntegerField(default=0)

quiz\_type = models.CharField(max\_length=10,default='rob')

quiz\_name = models.CharField(max\_length=20,default='rob')

correct\_questions = models.PositiveSmallIntegerField(default=0)

total\_questions = models.PositiveSmallIntegerField(default=0)

got\_socre = models.PositiveSmallIntegerField(default=0)

total\_socre = models.PositiveSmallIntegerField(default=0)

mouse\_clicks = models.PositiveSmallIntegerField(default=0)

pub\_date = models.DateTimeField('date published',auto\_now\_add=True)

#apps.py

from django.apps import AppConfig

class ExamsConfig(AppConfig):

name = 'exams'

#urls.py

from django.urls import path

from . import views

urlpatterns = [

path('', views.index, name='index'),

path('add-quize/', views.addQuiz, name='addQuiz'),

]

#views.py

from django.views.decorators.csrf import csrf\_exempt

from django.http import JsonResponse

import json

from .models import Question

# Create your views here.

def index(request):

print('')

return JsonResponse({

'status': 'ok',

'msg':'welcome'

})

@csrf\_exempt

def addQuiz(request):

if request.method == 'POST':

json\_data = json.loads(request.body.decode('utf-8'))

print(json\_data)

question = Question()

question.first\_name = json\_data.get('first\_name', None)

question.last\_name = json\_data.get('last\_name', None)

question.student\_id = json\_data.get('student\_id', None)

question.quiz\_type = json\_data.get('quiz\_type', None)

question.quiz\_name = json\_data.get('quiz\_name', None)

question.correct\_questions = json\_data.get('correct\_questions', None)

question.total\_questions = json\_data.get('total\_questions', None)

question.got\_socre = json\_data.get('got\_socre', None)

question.total\_socre = json\_data.get('total\_socre', None)

question.mouse\_moves = json\_data.get('mouse\_moves', None)

question.mouse\_clicks = json\_data.get('mouse\_clicks', None)

question.save()

print(question.first\_name)

res\_json={

'status':'ok',

'data':json\_data

}

return JsonResponse(res\_json)

#manage.py

#!/usr/bin/env python

import os

import sys

if \_\_name\_\_ == "\_\_main\_\_":

os.environ.setdefault("DJANGO\_SETTINGS\_MODULE", "mysite.settings")

try:

from django.core.management import execute\_from\_command\_line

except ImportError as exc:

raise ImportError(

"Couldn't import Django. Are you sure it's installed and "

"available on your PYTHONPATH environment variable? Did you "

"forget to activate a virtual environment?"

) from exc

execute\_from\_command\_line(sys.argv)