2018

Sprint 1

CITS3200 – GROUP B

MICHAEL BLEAKLEY, MINRUI LU, CHEN LIU, SAMUEL FAIRS, LACHLAN BUNNEY, JOSHUA MILAMBO

Scope of Work

Preface:

This document addresses the requirements of Exam Questions Database Web-app. The intended audience for this document are the group members, the client (John Brooke), the auditor (Daniel Cowen) and the Unit Coordinator (Michael Wise).

Target Audience:

Client, Group members, Auditors

Group-b Members:

Michael James Bleakley
Samuel Eric Lenagan Fairs
Lachlan Alexander Bunney
Chen Liu
Minrui Lu
Joshua Teodros Milambo

MILESTONES

- 20/08 Sprint1 Due→Documentations, front-end framework.
- 17/09 Sprint2 Due→Implementation of Server-side framework and database.
- 22/10 Sprint3 Due→Datum inserting, website integration, testing and user, maintenance guide.

1.0 General Goals

Our goal is to design a website server which stores exam questions, the usage information about each question and allow Users (mainly physics professors) to:

1. Browse, search (base on usage information, keyword), select and export the questions in the server as a zip file for preparation of their future exam paper.
2. Upload new questions and usage information to the server.

2.0 Current System

The current system used by our clients is store the questions in a word document, manually update the usage information in that document and pass the document manually between the professors.

3.0 Proposed System

3.1 Overview

Our system is a website server which allows the authorized users to upload, browse, search, select, export exam questions and associated information.

3.2 Functional Requirements

1. Client authentication; certain users have restricted accesses to specific questions.
2. Display questions and their usage information in a proper format.
3. Allow searching based on usage information and keywords of questions.
4. Export selected questions in a zip file.
5. Upload new questions and usage information.
6. Secure storage of exam questions.

3.3 Non-functional Requirements

1. Simple and clear user interface.
3. Downloadable instruction video showing how to use the website.

3.3.1 User Interface and Human Factors

1. Potential Client:
    a. The primary clients for the website will be the lecturers from physics faculty who search and export questions for the preparation of exam papers.
    b. Administration of physics faculty might also use the website to regularly clear up (like delete repeated questions) stored questions and associated information.

2. Design Ideology:

Because there might be users that are not as comfortable with the Internet (some senior lecturers), the UI will be designed in a simple, straightforward style with each page has a specific function. This makes the application intuitive and accessible for all users.

3. Protection:

The website will only be visited by authorized lecturers and administrators, so the integrity and professionality of the questions is not a concern.

The wrong operation of deleting questions will be a potential problem, so the deleting function will only be available to the authorized administrator from the physics faculty.

4. Devices:

The preparation of exam papers is a serious academic process, required to be done on a computer. For this reason the design of our web application will focus on serving computer browsers with large screen and enough processing capability to run AngularJS on the client-side.

3.3.2 Documentation

1. An acceptance test set with the potential audience:

a. Team members who use it to test errors, track progress and improve efficiency by analysing test results.

b. Auditor using it as a marking schema of our work.

2. Skill audit:

. Team members using it to assist our allocation of work.

a. Auditor using it as a marking schema of our work.

3. Risk register:

. Team members use it as a warning of the potential risk that needs to be addressed in the process of designing the website.

a. Subsequent administrator of the website using it as a guide to avoid the potential mismanagement of the system.

b. Auditor using it as a marking schema of our work.

4. Stories (at different stages):

. Team members using it for better time allocation, better understanding of the next stage, and analysing the completeness of work.

a. The initial client using it to check completeness at each delivery point.

b. Auditor using it as a marking schema of our work.

5. User instructions:

. For the client to understand how to use the web application.

a. Auditor using it as a marking schema of our work.

6. Administrator instructions:

. For the potential administrator of the web application to understand how the web server works and how to maintain and modify the database.

a. Auditor using it as a marking schema of our work.

7. Meeting minutes and booked hours spreadsheet:

. For team members to record their work and adjust their work ethic as required.

a. Protection against potential conflict of the intellectual property in the future.

b. Auditor using it as a marking schema of our work.

8. References:

. For all website users.

a. The reference sources.

3.3.3 Hardware Consideration

The server would be run on UWA hardware and the web application would be accessible from UWA shared computers. Considering the size of the physics question bank has not been confirmed by the client, memory size and auxiliary storage space are not known.

3.3.4 Performance Characteristics

The process time of all operations on the web application should be as short as a UWA official website (1-2 second response time). Considering the database schema, files are uploaded as MEDIUMBLOB files, therefore, the largest size of a file would be 16MB. Moreover, on account that the MySQL database we use has the maximum size of 256TB, capacity constraints do not need to be considered.

3.3.5 Error Handling and Extreme Conditions

Input error:
A current operation will be denied if there is an input error and a notice of that input error will be given.

Extreme conditions:

1. Excessive users.

a. Not very likely to happen as the web application is only used by professors from physics school.

2. Excessive questions length.

. Length of a question is constrained, and the action of storing oversized questions will be denied.

3. Excessive usage information.

. A HTML form constrains the length of the input.

4. Database Breaking

. A situation in which the Database fails to work or fundamentally breaks. This would have to be controlled / handled through maintaining a regular backup schedule and having people on hand who understand the inner workings of the database.

3.3.6 System Interfacing

The search input will be text, and it has to be input through the form on the web application presented in the navigation bar or on the index page.

The input of new questions has to be a zip file containing latex files and associated figures. This zip file won't be used for displaying questions and it will only be stored as a block in the database, so errors in this input file will not affect the functionality of web application.

The output of browsing questions will be an interactive PDF on the website.

The output of exporting questions will be latex zip file.

3.3.7 Quality Issues

Database security:
Using MySQL as the database is not a perfectly secured solution. However, the server will be coded in JavaScript which enables future change of the database to MS SQL or Oracle. This will be considered when the client has increasing demand for security.

Running time and inspection:
The web application will be online 24-7, and we suggest that it should be inspected each semester after delivered.

System failed:
If the web application failed, the server could be restarted in 10 minutes. The current input might be lost, but the database will not be affected.

Usability:
The web application can be run on all university servers as it is coded in HTML5 and javascript.

3.3.8 System Modifications

Database:
a. The size of the database will likely to be extended as the number of questions will continuously increasing.
b. The type of database is likely to be changed to improve security.

Editing function:
a. The further editing function of questions might be implemented

Sharing function:
a. The function of sharing questions by email might be implemented.

3.3.9 Physical Environment

The website will run on the server in the physics faculty, and it will mainly be accessed in the physics office and home, of lecturers and administration.

3.3.10 Security Issues

The access of data must be secured as it will be a potential exam question for coming exam.
The website and database will be stored on the server of physics where is highly secured.

3.3.11 Resource Issues

The IT team of UWA should be responsible for the installation, data backing up per semester, accessibility updating and regular maintenance.

3.4 Constraints

The website is programmed in html5, JavaScript Java, and SQL which can be run and access any browser.

Constrained by coding abilities, latex files and their associated figure files cannot be uploaded separately.

Constrained by code complexity and running performance, users cannot search random words in latex files but latex files' key words.

Temporarily, there are no legacy systems for this project.

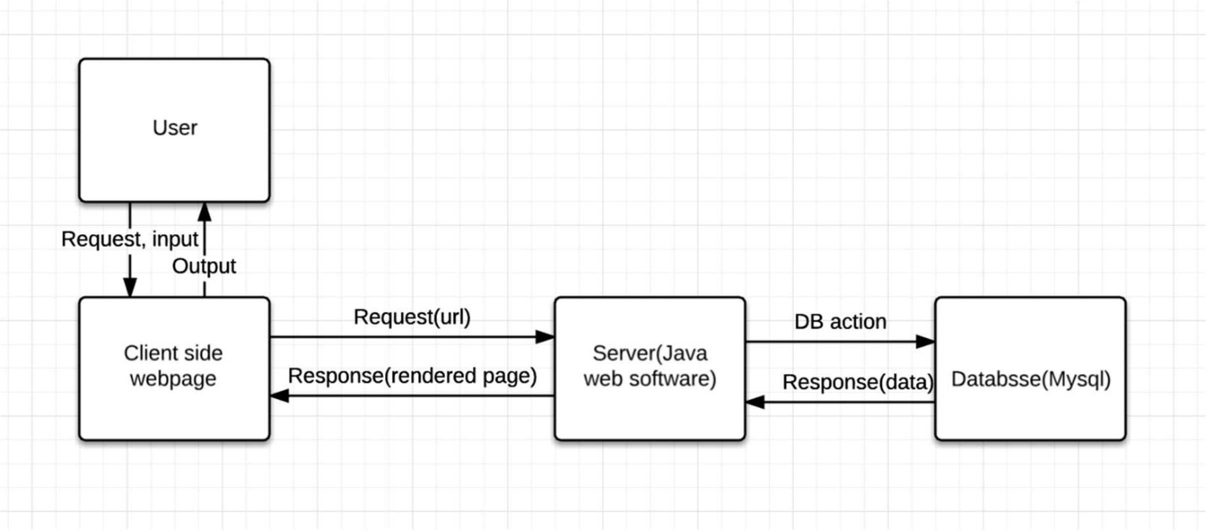
3.5 System Model

3.5.1 Scenarios, Use Case Models

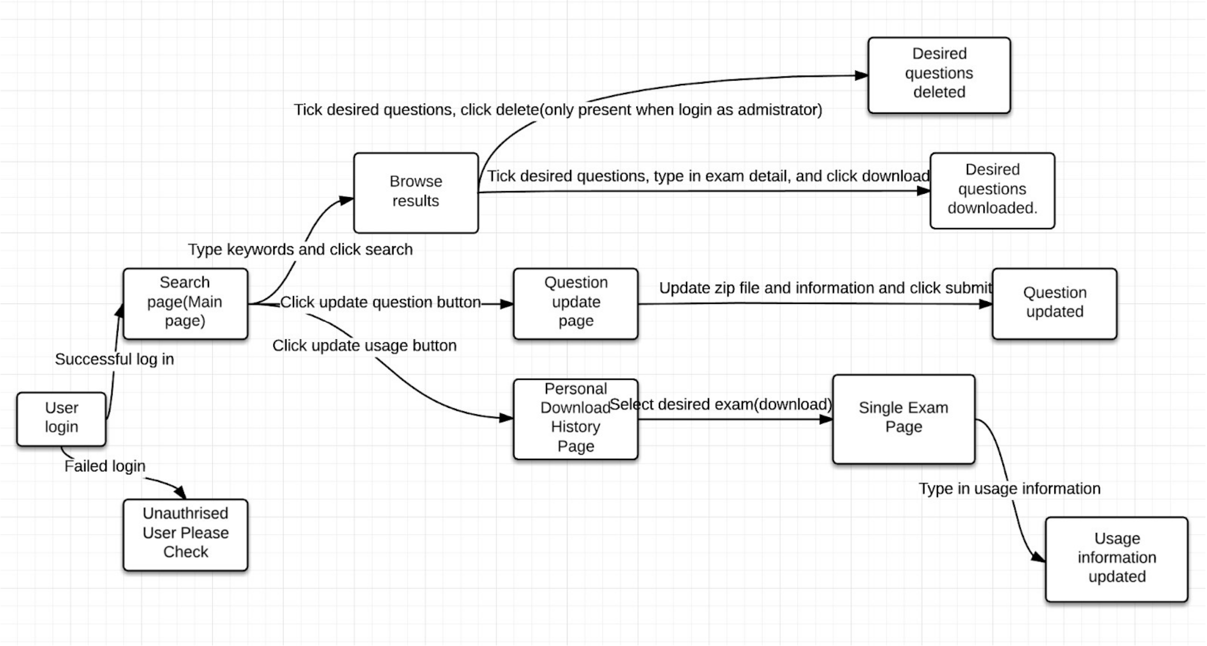
Download questions: User login→ Main page→ input keyword, browsing result, select questions and download selected questions.
Upload questions: User login→ Main page→ question upload page→ upload questions and key word of questions into database.
Update usage information: User login→ Main page→ usage update page→ update usage information of questions.

3.5.2 Use Case Models

3.5.2.1 Actors



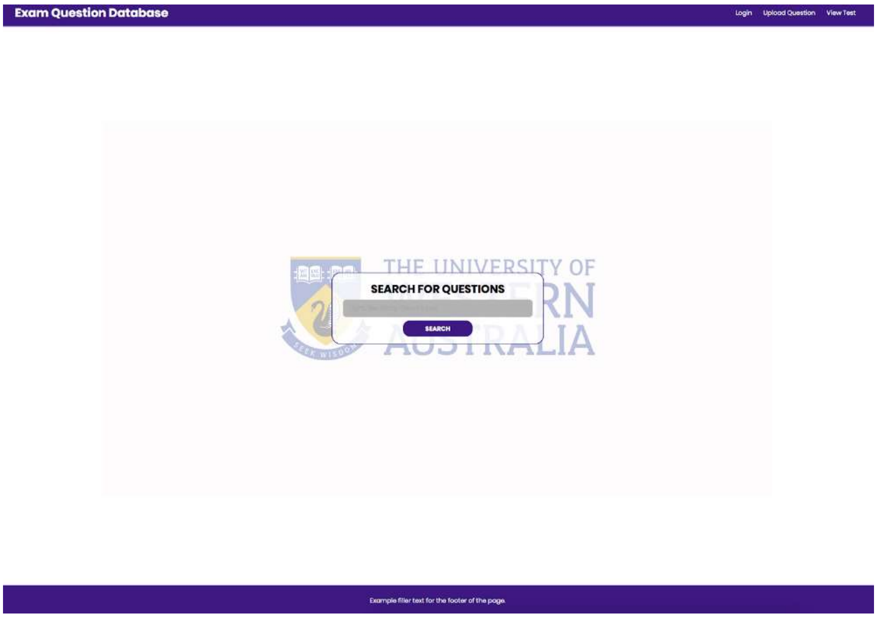
3.5.2.2 Use Cases


Desired questions deleted
Tick desired questions, click delete(only present when login as admistrator)
Browse results
Tick desired questions, type in exam detail, and click download
Desired questions downloaded.
Type keywords and click search
Search page(Main page)
Click update question button
Question update page
Update zip file and information and click submit
Question updated
Click update usage button
Successful log in
User login
Personal Download History Page
Select desired exam(download)
Single Exam Page
Failed login
Type in usage information
Unauthrised User Please Check
Usage information updated

3.5.3 Object Models

Our plan is to divide the project into three phases: Front-end web-page development, Java based server development and MySQL database development, and at different phases we will create and store data in different carrier: JavaScript to MySQL. At this stage, the stable object model is still unclear as our knowledge and requirement of data model will continuously grow and change overtime.

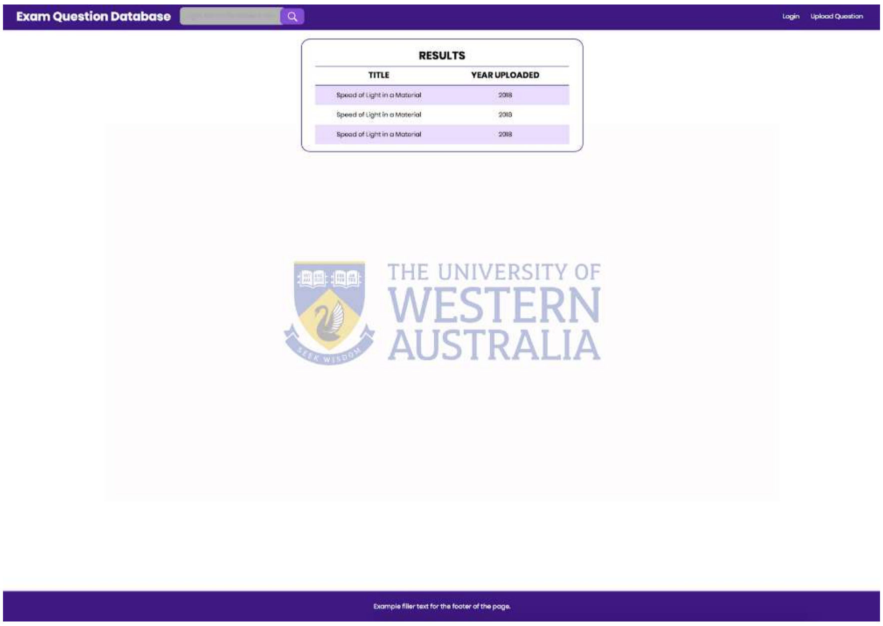
3.5.5 User Interface - Navigational Paths and Screen Mock-ups



Example filler text for the footer of the page.

Figure 3.1 Search Page

This page allow users to search questions or navigate to other pages from here.

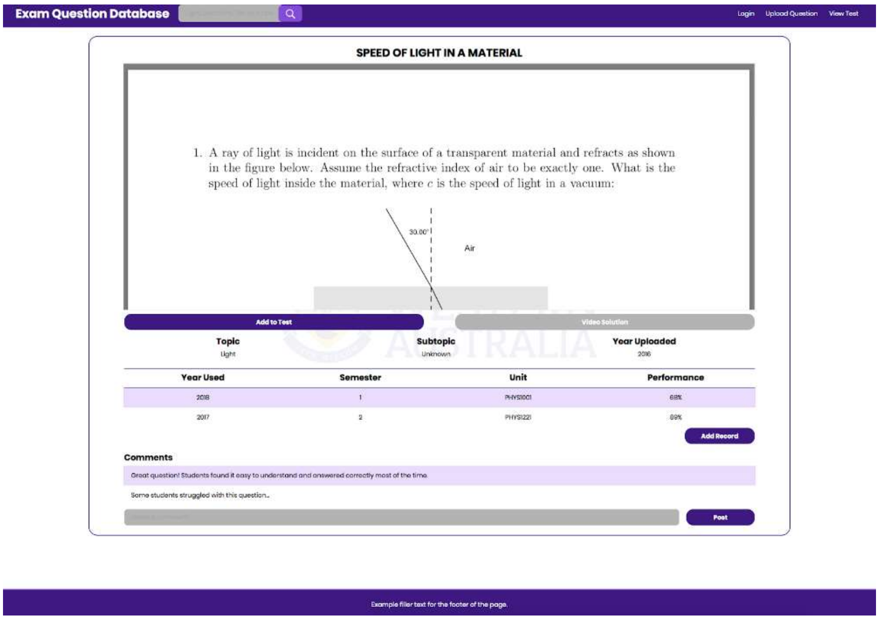


TITLE	YEAR UPLOADED
Speed of Light in a Material	2018
Speed of Light in a Material	2018
Speed of Light in a Material	2018

Example filler text for the footer of the page.

Figure 3.2 Result page

This page displays a table of questions that match with users’ search.



SPEED OF LIGHT IN A MATERIAL

1. A ray of light is incident on the surface of a transparent material and refracts as shown in the figure below. Assume the refractive index of air to be exactly one. What is the speed of light inside the material, where c is the speed of light in a vacuum:

Topic	Subtopic	Year Uploaded
Light	Unknown	2016

Year Used	Semester	Unit	Performance
2018	1	PHYS1001	68%
2017	2	PHYS1221	89%

Comments

Great question! Students found it easy to understand and answered correctly most of the time.

Some students struggled with this question...

Figure 3.3 Preview Page

This page displays the question and usage information that user want to preview.

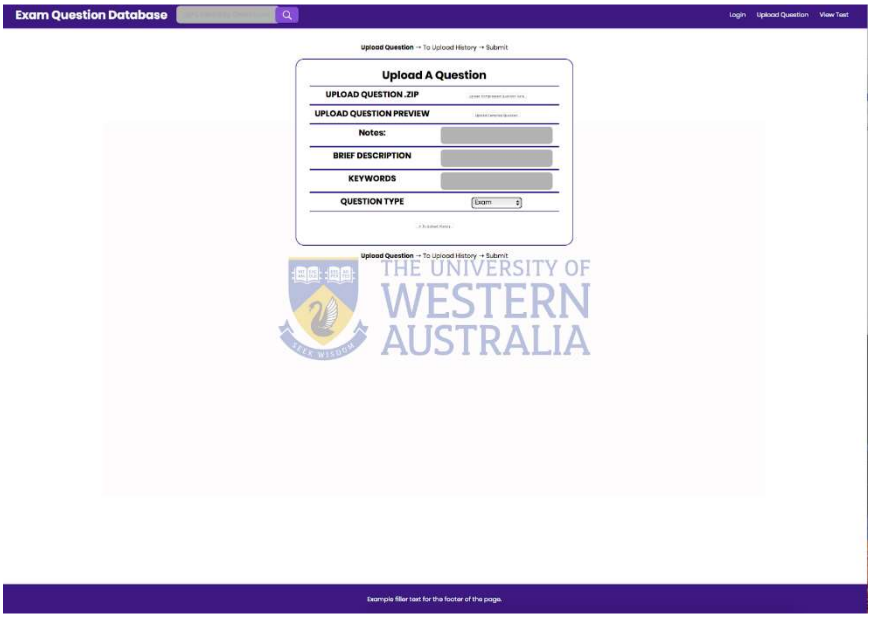
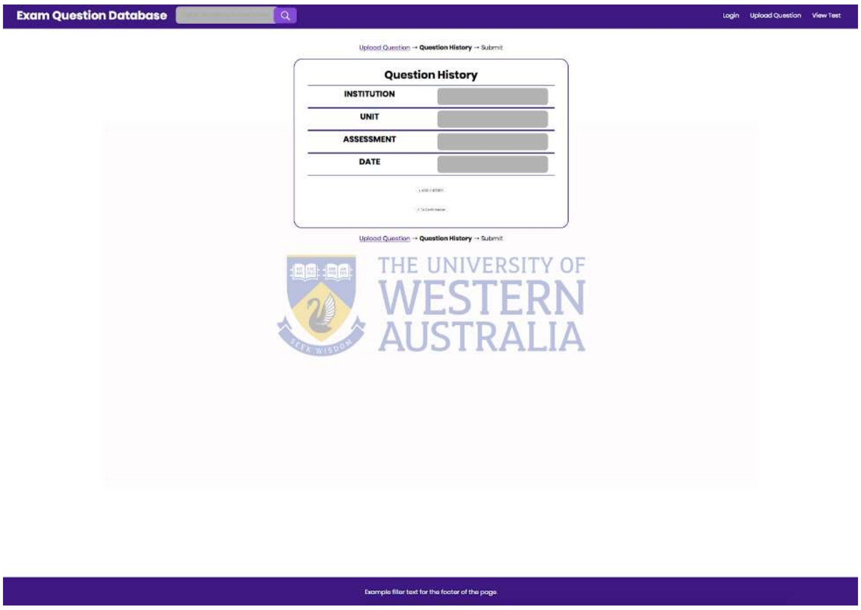


Figure 3.4 Uploading question page

This page allows users to upload new questions.

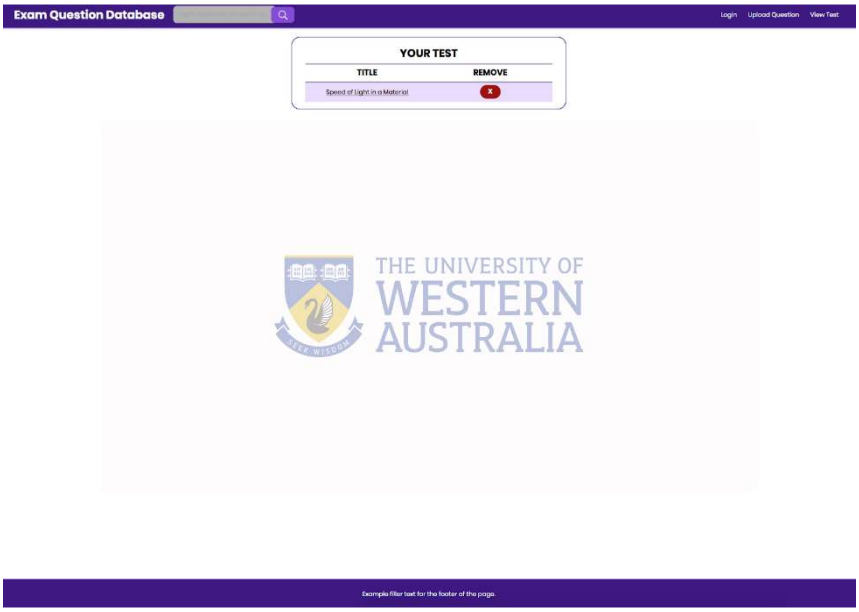

Exam Question Database
Login
Upload Question
View Test
Upload Question → Question History → Submit
Upload Question → Question History → Submit
THE UNIVERSITY OF WESTERN AUSTRALIA
Example filler text for the footer of the page.

Question History

INSTITUTION	
UNIT	
ASSESSMENT	
DATE	

Figure 3.5 Usage updating page

This page allows user to update usage information of given question.


Exam Question Database
Login
Upload Question
View Test
X
THE UNIVERSITY OF WESTERN AUSTRALIA
Example filler text for the footer of the page.

YOUR TEST

TITLE	REMOVE
Speed of Light in a Material	X

Figure 3.6 Download history page

This page allows user to browse his download history.

Project Acceptance Tests

Objectives

The Purpose of this aspect of Sprint 1 is to outline the various tests that the group will complete over the course of the project so that they can document progress and set themselves achievable and realistic goals.

The Completion of the following tests will also allow the group to determine the success of the project at various intervals and at its completion. As you will see, tests one and three relate specifically to the end product developed by the team whereas test two relates to how the team will view project success during the completion of the project.

Tests Summary

1. The Database System

a. This includes all relevant aspects such as the storage of documents and how they are accessed
b. Intactness of downloaded files.

2. Smaller Aspects to be incrementally Tested

. UI / Front End
a. Database
b.

3. Extra Work Done by the team on the system

. Due to the nature of the project there is a lot of room for the group to expand on the 'basic' formation of the system
a. This work will be done in accordance to what the client (in this case John Brookes) sees as the most valuable to the system
b. What the client values and what the group sees as achievable will be weighed up in this test

Testing Strategy

To conduct both formal and informal tests with the following conditions

- Who
  - Group Members
  - Client - John Brookes
  - Mentor - Ivan Zelina
- When
  - During Meetings with the client
  - When specific aspects of the project are "finished"
- Where
  - Room 461 of Physics (Conference Room)
  - CSSE Labs
  - Other Areas of Campus such as study rooms in the libraries

Test 1 - Major Functionality Test

Test Specification

This test will be a usability test completed by the client. In this test, the client will offer their opinions on the following

1. User Interface

2. Speed of Program
3. Output of the program
4. Ability to input files to the program
5. Other Elements of the program such as

a. History input
b. Search interface
c. Editing Exam Paper
d. Exam Paper Download

6. What aspects of “Extension” they want us to address first

. For example, the ability to upload more than one type of file or the ability for student access

Test Description

This Test would be conducted in a fairly unrestricted manner. The only emphasis that the team would place in the area of specific directions is making sure that the client accesses all areas of the program so that they can provide feedback on how the program works.

The main things that the team will have to provide to the client will be a username and password that allows them to access into the database system. Secondly, if the team notices that the client has missed a key aspect of the program or is misusing the program they should direct the client as they see fit.

Ideally, this test would be carried out in a room that has access to a projector so that the client can carry out the test on a large screen and members of the group can point out specific things (as needed) on the projected image rather than the screen of a small laptop. A room that would be suitable for such a test is the conference room in the physics building on the fourth floor (461) which has been used by the group previously to discuss other matters of the project with the client.

Finally, this aspect would be completed close to the end of sprint two or one week into sprint three. Setting this as a timeline would allow the group to act on any extra functionality that the client suggests or fix any bugs that come as a result of the test.

Test Analysis Report

The main analysis will be recorded in the minutes that relate to the meeting that this test is executed in. However, the main areas to be noted when assessing the client’s satisfaction and the completion of are:

1. Verbal Satisfaction

a. The group would have to specifically ask the client what they like, what they would change etc.

2. The Program working without fault in a real world use situation
3. The speed at which the program is realistically usable

. How fast is the search
a. How fast is the logic
b. Etc.

Test 2 - Small Functionality Tests

Test Specification

This test is a little different to the one previously described as it doesn't have as rigid a testing structure. The main goal of test two is to allow us to keep each other accountable and periodically test how the various aspects of the project are working together

Because our group has a very varied skill set, some members may not be able to physically code (or understand the development of) different aspects of the project. This means that they will have to have input in other ways such as providing various testing scenarios that the original coder may not have thought of.

Test Description

- Members Required:
    - Preferably 3 consisting of the primary coder and two other members who will oversee the testing
- Specific Aspect To be tested
    - Can include by not limited to, front end UI, zipping and unzipping of folders, importing files into the database.
- Testing Schedule
    - Due to the nature of this test various aspects should be ongoing. The tests shouldn't be only at the end of the development of a certain aspect of the project
    - Specifically, the developer should test the program incrementally as they've developing the program and then when they believe the program is finished they should bring in the other two members (assigned by the other members) and test the program again.
- Testing Procedure
    - As the program is being tested the tests that are being run should be outside the bounds of reasonable testing purposes. So the tests that are run are unrealistic and could never occur in a real world environment
    - By Testing in such a way we ensure that the program shouldn't break when the clients tests it as they would in test 1

Test Analysis Report

The Analysis report criteria of this test are

- Tests passed and failed
    - Should be 100% pass ratio
- Speed of Tests
    - Tests should run in a prompt manner
    - This is especially important for search related tasks

Test 3 - Extra Functionality Tests

Test Specification

This test is very similar to how the small functionality tests would occur. The description of this part is almost identical to that of Test two. The major difference of this test is that it occurring specifically depends on the completion of Test 1. Also, the specifics of how the test will be run are dependent on the extra functionality that the client wanted

The major distinctions of how this testing will be run is that the whole group will most likely be involved in the development of these areas due to the fact that when they are completed when everything else is completed. What this will mean is that the group will have multiple people working on it and therefore more people to help with testing (unlike the 3 as depicted in test two).

Finally, this test will also include a similar testing aspect to test one as when the group completes the extra work they will have to present it to the client in a similar way.

N.B For the description and analysis reports of this test refer to the tests two and one in that order

Skill and Resources Audit

Identified Technical Skills	Description	Why this skill is necessary	How we can learn/improve this skill set
LaTeX	LaTeX is a system used for document preparation on medium to large documents.	The input of physics questions fed into the project interface are of a .tex format (Latex file) or a .txt file format. A sound understanding of the compilation and storage of these files will be necessary to successfully use them.	• Online research • Listening to our Mentor • Learn from peers at University
Html	Html is used for creating web pages and applications.	We will need to know this language to make the online interface.	
JavaScript	JavaScript is an object-oriented language used to make web pages more interactive and allows more complex functions on the client side of an interface.	This will be used for client-side scripting, including necessary pre-processing functions and making a more dynamic interface.	
MYSQL	MYSQL is a database management system.	We need to know this language to create a secure place to store and access the physics questions. The more we know about the language, the better we can design the databases to improve efficiency.	
CSS	CSS is used to alter the presentation of web pages.	The front end of the project might require a certain aesthetic that would be very difficult to achieve without CSS. In this case it would require an interface like the standard UWA webpage.	
Node.js	Node.js is an open source server environment that is run on JavaScript.	This will be used to make the web server.	

Identified Non-Technical Skills	Why this Skill is necessary	How we can learn/improve this skill set
Communication	With a large project comes a significant workload. Clear communication allows effective delegation of tasks and progress reports of those tasks. It also allows disagreements and disputes to be dealt with faster.	• Slack • GitHub • Google Drive • Facebook Messenger
Flexibility/Adaptability	Although a general understanding of how to complete the project may be reached, there is always a chance that circumstances may change. Being adaptive will be necessary in a situation like this will be crucial to the completion of the project.	• Accept multiple perspectives • Listen • Get feedback
Time Management/Organisation	Time management will be a large consideration during the entire project. Using time effectively and being able to correctly allocate time based on the importance of the task will be crucial. It will help prevent the same work being done repeatedly or work not being done at all.	• Have a schedule • Have goals • Allocate time based on importance of task
Cooperation	Cooperation is a necessary trait for a group project because without working together the workload will be far too great. Cooperation will be especially important when working on the same pieces of code and when critiquing work. It is important to understand that we are working together and not against each other.	• Slack • GitHub • Google Drive • Facebook messenger
Problem solving	There will be many large and small problems that need solving within the project. Without the ability to think of creative solutions to these problems the progress of the project will come to a halt, potentially wasting valuable time.	• Online research • Listening to our Mentor • Learn from peers at university

Resources	**Description**	**Why this resource is valuable**
Internet	A global computer network, the internet is the home of vast amounts of information on all topics.	We can find appropriate information on topics of interest and use it to communicate effectively.
Mentor	For this project we were assigned a mentor with project experience to oversee our work and provide us with advice.	Working someone who has decades of project experience will help provide more perspective on how to approach design and implementation..
University Staff and Students	UWA is filled with knowledgeable students and staff that would be willing to help provide thoughts and information about a project.	Like the mentor, we can gain more perspective and information about our project.
Books	Books have always been a reliable source of information, being comprehensive and detailed.	If we require very specific information about a topic, finding a book might be more efficient than looking online.

Risk Matrix

Identified Risks	Risk Score				Comments	How To Control	Residual Risk (Risk After Control Applied)			
	C	L	E	R			C	L	E	R
Loss Of Data	5	2	5	50	This could include the but isn't limited too • Accidental Deletion of important files • Erasing of Files on GitHub • Accidental 'corruption' of files on GitHub	This can be controlled by the members of the group (1-all) periodically downloading the registry on GitHub and all other Files Doing this will create a secondary/tertiary source of data.	5	1	2	10
Data Security	5	3	4	60	This is the security of the data that is stored in the database. This will primarily be an issue when trying to rollout the database If proper encryption isn't applied at this stage there is a potential that data can be corrupted, stolen or deleted entirely by an attacker	Obviously applying proper encryption on all data in and out of the database will be crucial but can also inhibit the amount of chances that an attacker would have to effect this. What we will do is test all connections (prior to encryption especially) on single machines or on a closed network in which we can control the devices that have access to it	5	1	1	5
Websites Being Down	3	1	5	15	This is extremely bad for a large group assignment and also it would affect the group heavily if websites where we have to submit parts of the assignment. For example, if GitHub was down on the day in which sprint 1 was due the group would have no (clear) way to submit the assignment.	To prevent this being an issue, in a similar way to the control of loss of data the group can create redundancies such as backups of data. Also, the group should be aware of the status of all required technologies so in the event of a website going down, the group is able to fix the issue as so as possible	2	1	3	6

Group Members Getting Sick (or other like situations)	2	3	4	21	In a large group assignment that exists over the course of a semester it is almost inevitable that a member of the group will get sick. If this occurs the group will be placed under the strain of not having a member being able to work and (potentially) lacking a particular set of skills that are integral to the assignments completion	Avoiding getting sick is a lot harder to do than planning for the event, so that is a better way to control this risk. The group should delegate tasks in such a way that multiple people are responsible for the task so should one of them get sick the task isn't completely disregarded. Also, the group members should (hopefully) have some knowledge of how all parts of the project work so that any discrepancies caused by the loss of a member can be managed.	1	3	3	9
Lack of group cohesion	4	3	4	48	In such a large group there is a tendency that people will start to butt heads or disagree on how they think the group should proceed with the project. This can cause members of the group to grow disinterested in the project and become a hindrance on the rest of the group when they aren't working at their full capacity.	To avoid this the group has to be careful to avoid the quashing of ideas from all members of the group. A good strategy to avoid this is for (in meetings) the group to create an agenda for discussions in which everyone has their own time to speak without interruption. What this does is allows people who are less controlling in conversations to still be heard. Also, the noting of this down in minutes will allow all members ideas to be recorded and potentially used.	2	2	4	16

Matrix Key

C = Consequences (1-5): 5 being the worst

E = Exposure (1-5): 5 being the most people/things exposed to this

L = likelihood of occurrence (1-5): 5 being incredibly likely

R = Total Risk (multiplication of the other 3 values)

Stories

Preface:

This document addresses the stories of Exam Questions Database Web-app in Scrum agile methodologies. The intended audience for this document are the group members, the client (John Brooke), the auditor (Daniel Cowen) and the Unit Coordinator (Michael Wise).

Target Audience:

Client, Group members, Auditors

Group-b Members:

Michael James Bleakley
Samuel Eric Lenagan Fairs
Lachlan Alexander Bunney
Chen Liu
Minrui Lu
Joshua Teodros Milambo

Client:

John Brooke

1.0 General Goals

Our goal is to design a website server which stores exam questions, the usage information about each question and allow Users (mainly physics professors) to:

1. Browse, search (base on usage information, keywords), select and export the questions to the user as a zip file for preparation of their future exam paper.
2. Upload new questions and their usage information to the server.
3. Upon downloading for a formal use allow professors to update question histories.

2.0 Themes

Note:

- In the project, users are lecturers from physics faculty who are going to make test papers.
- Stories are listed from higher value to lower value under each epic.

2.1 Theme 1 Question Uploading Webpage

2.1.1 Epic: Uploading Latex Files and Their Preview Images

· As a user, I want to upload a question in a zip file in which contains Latex file and associated figure images.
· As a user, I want to have a button to upload zip files.
· As a user, I want uploaded questions can be safely stored in database and not collide with other questions.

2.1.2 Epic: Adding Information of Uploaded Question

· As a user, I want to add keywords for uploaded question, so that other users can get the question by searching the key words.
· As a user, I want to have a text box to type in notes of uploaded question, so that other users can get more information of this question.
· As a user, I want to have a text box to type in short description (usually the first sentence of question), so that other users can have an explicit recognition of the question.
. As a user I want to be able to record the history of usage of each question when I upload it

2.2 Theme 2 Exam Paper Editing Webpage

2.2.1 Epic: Exam Paper Registration

· As a user, I want to have four text boxes to register the new exam information of Institution, Unit, Assessment, and Date.
· As a user, I want to have a text box for additional notes of new exam paper.

2.2.2 Epic: Searching Questions

· As a user, I want to have a search box, so that I can get the wanted questions by searching keywords or other question information.
· As a user, I want to see on the searching result interface, each matched result is along with the question's short description, a button for previewing the question, a button for viewing question history, and a button for adding a question to the cart.
· As a user, I want to see information of institution, unit, assessment usage, date, result statistics, and notes after clicking the history button.

2.2.3 Epic: Question Cart

· As a user, I expect to see for each line, there would be the short description of each question, a button for previewing, a button for viewing history, and a button for removal from cart.

- As a user, I want to have a submit button at the bottom of question cart, so that once click on it, I can download selected questions and corresponding history would be added to database.
- As a user, I want to have an access to change the question order in the exam paper.
- As a user, I expect when I compile all questions into an exam paper, the reference of a questions image should not conflict with an image relating to a different question of each question's images would not conflict each other.
- As a user, I want to save my temporarily chosen questions in my account background, so that I can continue editing my exam paper in the future.

2.3 Theme 3 Result Updating Webpage

2.3.1 Epic: Search Box

- As a user, I want to have three dropdown-boxes corresponding to institution, unit, and assessment & date, so that I can filter the targeted exam paper.
- As a user, I want the dropdown-box can be up to date with database.

2.3.2 Epic: Updating Interface

- As a user, I expect on the update interface, each line represents one question, with four columns for the short question description, a button for previewing, a text box for inputting the number of correct students, and a text box for inputting the total number of students.
- As a user, I want that after all results are uploaded, I can have a preview of the whole paper result, with institution, unit, assessment, date, and correctness statistics on each line for each respective question.

3.0 Intermediate Acceptance Criteria for Sprint 2

3.1 Question Uploading Webpage

- Uploaded zip files are stored in database logically. There is no collision when extracting files.
- There is no data loss in the uploading process.

3.2 Exam Paper Editing Webpage

- After searching for keywords, matched questions' short description, preview and history can be shown.
- After clicking add button, the targeted question is added to question cart successfully.

· After clicking the submit button, questions in the question cart are exported correctly into a .zip file together

3.3 Result Updating Webpage

· For each dropdown box, all available choices in database can be shown.
· After clicking dropdown box options, SQL queries return the wanted results, and webpage displays them correctly.
· After updating the exam result, corresponding data are inserted into database correctly.