

Helen McDonagh 3rd year project

StudyBuddy

G00320304 | Computing in Software Development | 24th of April 2017

# Table of Content

Introduction

Project Idea and Development

Technology Used

User Guide

Development Methodology

Limitations and Known Bugs

Conclusion

References

# Introduction

My intention for my final year project was to create a Note Storing System for secondary level students so they could have all their notes all in one place, I want to use eclipse to make a JSF style project written in java with a sql database to store all the data. I had intended to allow students add images, diagrams and to search through the database for their notes. There is a list of all subjects that the student can use and then develop the notes from then on.

I have some understanding of JSF but I think I should develop it a bit more to do my project. I am used to java is has been one of my core languages for the last three years, and SQL. This project has helped me develop SQL queries and how to build a database. The database that I am using I created myself, and it will be stored locally on the machine. I am using apache tomcat for a server to show the application on localhost.

The information will be stored in a SQL database and kept, the database is the key factor for this application. I wanted to make this application because I have seen that some students having problems writing down their notes losing vital sheets of paper. So, I thought why not make it portal and paperless, so all they would have to do is turn on their computers and run it and have all of it in front of them.

When I finish this project, I expect to have better understanding of JSF and SQL.

# Project Idea and Development

My idea for this project came from seeing and experiencing the problem with studying and writing out notes and losing paper, mainly not being organized enough. I asked my Supervisor about this idea, within a meeting we designed the structure of the database and how everything should be link throughout the tables within the database. Within a couple of days, I created the database and named it studybuddy.

I started looking up the best way of going about developing the project, I spent a week deciding how to structure of the java code and layout of the displayed pages. I create a few basic pages to start off to read in the data from the database and display them in a table. Once I had them displaying data, I created another page to show links for exam papers, resources to help students get notes and to learn through different forms, like using videos from National Geographic and History channel. Once the pages started to take shape, they didn’t look up to standard so I decided to add Bootstrap to the project to help with styling for a more professional look. I then created a page to display images and to search for all notes assigned to one subject but unfortunately they didn’t go to well. I began on creating pages to adding, updating and deleting notes from the database. These pages worked out very nicely, they are efficient.

I designed this project as a single-user system for one user at a time, all they need to do is look up for the subject they want to add notes for, then add a subject topic id and then add their notes. After that they can read their notes in numerical order ascending by subject id. There is a page on that has resources to help the student find exam papers, other sources of information on all junior and leaving certificate subjects, history channel and national geographic to help them learn through forms of media.

The application produces a webpage in eclipse, which then you can copy the link and paste it into any web browser. It will the show you the homepage which then allows you to navigate throughout the other pages. This project will not need any static or dynamic software updates for any aspects of it. This project is to help students with their studies and help them concentrate.

# Technology Used

The technology I used in my project are the following:

* Java\ Eclipse
* SQL \ wampserver
* JSF

***Eclipse***

Is defined as a platform that has been designed for building integrated web and application development tooling. The design of the platform doesn’t have a lot for the user in functionality and design, the overall platform promotes rapid development of integrated system features based upon plugins. It delivers a UI/ user interface mode with working tools. It can handle and run numerous OS/ operating systems despite the fact offering a strong integration operating system. Eclipse offers multiple plugins and APIs which are supported on its operating system. Eclipse has a very dynamic architecture structure for discovery, loading and running of their plugins. The platform handles the logistics of finding and running the right code. The platform UI provides a standard user navigation model. Each individual plugin can concentrate on their appointed task(s). Their task are mainly testing, animating, publishing, compiling, debugging and much more.

***Java***

Java is a general purpose, high level programming language developed by Sun Microsystems. It was initially started back in 1991 and originally called OAK, at the time it was created it was designed to handheld devices. It was developed in 1995 for the budding new World Wide Web\Internet and it changed its name to java.

Is defined as an Object-Oriented Language and it is very like C++, the source code files with a *.java* extension are compiled into a format called bytecodefiles with a *.class* extension.

***Wampserver***

It is a web development platform on windows that allows you to make dynamic web applications with Apache, PHP and of course mysql. The wampserver automatically installs everything you need to develop databases and web applications.

***MySQL***

Is most popular and in demand open source SQL database management systems and was acquired by Oracle back in January 2010 which is developed, distributed and supported by their corporation now. MySQL is a relational database which stores data in separate table instead of placing all the data in one big unordered table. Its database structures are organized to optimize for speed in the physical layer. The logical layer deals with tables, views and rows and columns. Also, it’s software is open sourced. The MySQL Database Server is very fast, reliable, scalable, and easy to use.

# User Guide

***Installation***

First, download WampServer(<http://www.wampserver.com/en/>) from the website. It will contain new version of apache, php and mysql. Then wait for it to finish downloading. Then click on the executable file and run it, follow the setup instructions and wait. Once it has finished installing start wamp, a small little icon W will appear in lower right hand side of the toolbar. Wait for it to turn green, left click it and select MySQL\MySQL console. Open in notepad StudyBuddy.sql file in the GitHub repository, copy all the text and paste it into the console command line and press enter. Now once you have that done, go to <https://tomcat.apache.org/download-90.cgi> and download the 32-bit/64-bit Windows Service Installer, 9.0.0.M10. Select type of Installation Normal, click Next, click next on the Configuration Options screen, set the JRE (Java Runtime Environment) Path e.g. C:\Program Files\Java\jre1.8.0\_101 and click next and finally click Install.

Finally, now install Eclipse. Download and unzip Eclipse IDE for Java EE Developers from <https://eclipse.org/downloads/eclipse-packages/>, unzip the file just downloaded and click on the executable file. Follow the instructions and agree to the terms and conditions. Once it finishes installing, open eclipse and click on file and import project, select the final year project you downloaded from the GitHub repository.

***Starting Application***

When everything is installed, imported and created correctly you can run the application. So, first you must have the WampServer started running in the background to run project. Go to Eclipse, finalYear project, now there is two ways of starting this project:

1. You can open the WebContent Folder and right click on the Index.xhtml and run it on the Server. Copy and paste this URL localhost:8080/finalYear/faces/Index.xhtml into a web browser of your choice.
2. Or you can right click the StudyBuddy.war file and run it on the server, then go to the browser and paste the URL.

***How to Use Project***

**Homepage**

The homepage is the navigation tool to go through the main pages of the project. Here are the mains of the Application.

Homepage image

**Subject Listing**

Displays all the subjects that are stored in the database.

Subject image

**Notes**

The Notes page displays all the notes in the database and they are ordered in numerical order by Subject ID. You can navigate through the pages here to add Subject topic id, notes, update and delete notes as well. Click on Add Subject Topic to begin.

Notes image

**Add Subject Topic ID**

First insert the subject id you wish to assign to subject, then create topic id. Then Click add, now move onto add Notes but remember the topic id you created.

Add Subject Topic ID image

**Add Notes**

Now re-enter the topic id you created in the previous page. Enter the name of the topic and content of the topic you wish to create and click add. Now return to the Notes page and it has been added to the table in the database.

Add Notes image

**Update Notes**

To update your notes, you must look for the topic id of which note you wish to change. Enter the selected id, you can now change the name and details of the topic. When your finished all you have to do is execute it by clicking update.

Update Notes image

**Delete Notes**

Deleting notes are very easy, find the topic id of which note\topic you wish to delete. Enter the id and execute it by clicking delete, now return to your notes its gone, never to be seen again.

Delete Notes image

**More Information**

This page contains links to the examinations website which has all the exam papers for every subject in the curriculum for both junior and leaving certificate students. StudyNote has a lot of information on a huge number of common subjects for both levels. The History channel and National Geographic have an enormous amount of information in written and video Documentaries.

More Info image

***Database Design***

I used a wampserver to create the studybuddy database, I then create three tables and then I called them “subjects”, “topics”, “subject\_topics”. Here are the sql commands I used to create the database and tables.

Create Database ‘studybuddy’;

Create table if not exists ‘subjects’ (id` int(3) not null,`name` text not null,primary key (`id`)) ENGINE=InnoDB;

Create table if not exists `subject\_topics` ( `subjectid` int(3) not null, `topicid` int(11) not null, primary key (`subjectid`,`topicid`), key `topicid` (`topicid`)) ENGINE=InnoDB;

Create table if not exists `topics` ( `topicid` int(11) not null, `topicname` text not null, `details` mediumtext not null, primary key (`topicid`), key `topicid` (`topicid`))ENGINE=InnoDB;

I connected the subjects table by id to the subject\_topics table by subjectid. Then I connected subject\_topics table by topicID to the topics table by topicID. This connects the tables so when I create a new topic, I create a row in the subject\_topics table first and then add my notes in the topics table. So when I want to select all the English notes for example I create a join that will select everything from all tables and display all the notes under the subject english.

The database overall is quite simply design so anyone can read it and understand what it’s actually purpose is. I choose to use mysql because I understand how it works.

# Development Methodology

Throughout this project I used Oracle MySQL documentation to help me figure out the best way to connect the tables in the database together. I wanted to make this project into Graphical User Interface\GUI application at the beginning but a JSF web application was a more logical and cleaner design to use.

<https://dev.mysql.com/doc/refman/5.7/en/introduction.html>

# Limitations and Known Bugs

There are no major disadvantages to using JSF, it is very easy to style and it supports html, html5 but everyone still uses xhtml. Xhtml is a hypertext mark-up language and is part of the xml family. The projects minor limitation is that sometimes it is very slow to run and cuts out.

# Conclusion (Big Section)

I wanted to test myself,

What\why I decide to make first,

What didn’t work

What I learned

What would you have changed about the project

What its purpose is-market place, is it ready, study/learning tool

Why I built it the way I did

# References

1. <https://dev.mysql.com/doc/refman/5.7/en/introduction.html>
2. <https://eclipse.org/downloads/eclipse-packages/>
3. <https://tomcat.apache.org/download-90.cgi>
4. <http://www.wampserver.com/en/>
5. <http://www.oracle.com/technetwork/articles/java/javaserverfaces-135231.html>
6. <https://www.tutorialspoint.com/jsf/>
7. <https://www.w3schools.com/bootstrap/>
8. <https://www.w3schools.com/sql/DEfaULT.asP>