SCHOOL OF SCIENCE AND ENGINEERING

AUI EXCHANGE PROGRAM COURSE
MANAGEMENT SYSTEM

Capstone Project

April 2017

Prepared by:

Amine El Kaissi

Supervised by:

Dr. Naeem Nisar Sheikh
Dr. Sedki Samadi
Dr. Abdelkrim Ouardaoui
ACKNOWLEDGEMENTS

Acknowledgements are put at the very start of a project report for a reason: being grateful and thankful to the people who have helped you get back up when knocked on the ground. This is for the people that have participated in the shaping of this project.

To my mother, my father, my little sister and my little brother, I am so happy to have you by my side and providing me with the love and support that I need even with the physical distance that separates us, thank you and I love you.

I am very thankful to Dr. Naeem Sheikh for his guidance and motivation, to Dr. Sedki Samadi for his encouragement and support and to Dr. Abdelkrim Ouardaoui for his assistance and clarification of the questions that the application is supposed to answer.
# Table of contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACKNOWLEDGEMENTS</td>
<td>1</td>
</tr>
<tr>
<td>TABLE OF FIGURES</td>
<td>3</td>
</tr>
<tr>
<td>ABSTRACT</td>
<td>4</td>
</tr>
<tr>
<td>INTRODUCTION</td>
<td>5</td>
</tr>
<tr>
<td>Context</td>
<td>5</td>
</tr>
<tr>
<td>Scope of the project</td>
<td>5</td>
</tr>
<tr>
<td>Feasibility study</td>
<td>7</td>
</tr>
<tr>
<td>Economical aspect</td>
<td>7</td>
</tr>
<tr>
<td>Technological aspect</td>
<td>7</td>
</tr>
<tr>
<td>Technical aspect</td>
<td>7</td>
</tr>
<tr>
<td>STEEPLE analysis</td>
<td>8</td>
</tr>
<tr>
<td>METHODOLOGY</td>
<td>9</td>
</tr>
<tr>
<td>REQUIREMENT GATHERING</td>
<td>11</td>
</tr>
<tr>
<td>Functional requirements:</td>
<td>11</td>
</tr>
<tr>
<td>Non-functional requirements:</td>
<td>12</td>
</tr>
<tr>
<td>TECHNOLOGY ENABLERS</td>
<td>13</td>
</tr>
<tr>
<td>DESIGN AND IMPLEMENTATION</td>
<td></td>
</tr>
<tr>
<td>Model</td>
<td>15</td>
</tr>
<tr>
<td>Entity Relationship Diagram</td>
<td>16</td>
</tr>
<tr>
<td>Business Logic</td>
<td>17</td>
</tr>
<tr>
<td>Interface</td>
<td>19</td>
</tr>
<tr>
<td>CONCLUSION</td>
<td>29</td>
</tr>
<tr>
<td>WEBSITES</td>
<td>31</td>
</tr>
<tr>
<td>APPENDICES</td>
<td>31</td>
</tr>
</tbody>
</table>
Table of figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>adopted software engineering model</td>
<td>15</td>
</tr>
<tr>
<td>2</td>
<td>ERD used for the database</td>
<td>16</td>
</tr>
<tr>
<td>3</td>
<td>SQL statement behind the selection of the courses</td>
<td>17</td>
</tr>
<tr>
<td>4</td>
<td>SQL statement behind the deletion of courses</td>
<td>17</td>
</tr>
<tr>
<td>5</td>
<td>SQL statement behind the updating of courses</td>
<td>18</td>
</tr>
<tr>
<td>6</td>
<td>SQL statement behind the addition of the courses</td>
<td>18</td>
</tr>
<tr>
<td>7</td>
<td>view of the interface from the administrator’s perspective</td>
<td>19</td>
</tr>
<tr>
<td>8</td>
<td>combo box showing the list of universities in the database</td>
<td>20</td>
</tr>
<tr>
<td>9</td>
<td>combo box showing the codes of the courses in AUI</td>
<td>21</td>
</tr>
<tr>
<td>10</td>
<td>zoom-in on the grid of courses</td>
<td>22</td>
</tr>
<tr>
<td>11</td>
<td>zoom-in on the results based on the user’s selection</td>
<td>23</td>
</tr>
<tr>
<td>12</td>
<td>zoom-in on the results depending on the user’s new selection</td>
<td>24</td>
</tr>
<tr>
<td>13</td>
<td>zoom-in on the table after clicking on the ‘Reset’ button</td>
<td>25</td>
</tr>
<tr>
<td>14</td>
<td>zoom-in on the row being edited</td>
<td>26</td>
</tr>
<tr>
<td>15</td>
<td>fields that get filled to add a course</td>
<td>27</td>
</tr>
<tr>
<td>16</td>
<td>view of the interface from the student’s perspective</td>
<td>28</td>
</tr>
</tbody>
</table>
ABSTRACT

Automation is an important addition to today’s manual work and facilitates the work of many. Businesses and companies are migrating their data and tasks to a more dematerialized and intelligent way by the use of desktop and web applications that are linked to a database: this is exactly what this project aims at upon its completion. The transformation of a large Microsoft (MS) Excel sheet into a user-friendly web application would facilitate the searching and adding of courses offered abroad for both the student and the office of international programs (OIP) making the tasks much easier to accomplish and saving time and effort on both sides.
INTRODUCTION

Context

The monitoring of students of Al Akhawayn University (AUI) that go to study abroad for a semester or more has been done by the OIP for both the school of business administration and the school of humanities and social sciences and by Dr. Abdelkrim Ouardaoui for the school of science and engineering: they are in charge of inserting manually into an MS Excel sheet the details of each and every student including their full name, the institution in which they are studying, the courses they have taken, and more. The aim of this project is to ease up the process for both the monitors and the students by deploying a web application that facilitates the tracking of students and approval of courses from a side and the searching and adding of courses from another.

Scope of the project

The object of this capstone is to develop a user-friendly web application system which allows its user to manage information about courses offered by other universities. The application shall be directly connected to a database that is filled with all of the courses offered by the network of universities that are part of the bilateral and/or ISEP exchange programs. It shall be developed using C# as the language of implementation along with SQL for the database-related requests. Ultimately, the user of the application will be able to manage the approval of the courses or, in other words, decide whether course X at university Y complies with AUI’s equivalent course Z
or not, update the database of courses should the need arise, and search for specific courses with
the use of an advanced yet easy-to-use search tool.

By eliminating a lot of manual work and time that both Dr. Ouardaoui and the OIP spend on
managing this information currently without a system like this, this application aims to increase
productivity and speed up the feedback to the students about various universities and courses
they are considering for their exchange program.
STUDY

Feasibility study

*Economical aspect*

The cost of development of the project will be minimal as most, if not all, of the software used is freeware.

*Technological aspect*

Upon completion, the project will be tested locally while making sure that it retains its compatibility with the deployment solutions adopted by the university for final release through Jenzabar. It shall be developed using C# as the language of implementation along with SQL for the database-related requests.

*Technical aspect*

There are similar up and running systems that are used by other universities which provide a set of functionalities that are close to the ones wished for to be integrated into the project making this capstone technically feasible.
STEEPLE analysis

Social
The web application would facilitate the use of the database of courses and/or students saving up a lot of time and effort for its user.

Technological
There would be no need to enter, delete and/or modify anything manually from an entry: everything is automated and simplified with only a few clicks.

Economic
No cost whatsoever is to be recorded for this project as all tools that are used are freeware.

Environmental
There is no negative impact on the environment at all.

Political
No politics are involved in the creation and maintenance of this application.

Legal
Everything is perfectly legal and under the agreement of the supervisors and the university.

Ethical
There will be two types of users: administrators and students, both categories having a special set of tasks that they are allowed to do.
METHODOLOGY

The methods employed in order to tackle the project that I was invited to work on are the fruits of the computer science background that I have learned throughout my stay at Al Akhawayn University. Because the project is a web-based application, a software engineering approach was crucial in order to assure the health and success of the final product.

I have debuted the journey by sitting with my supervisors and discussing the main goals to be achieved through this capstone project. We came up with a list of requirements that were to be satisfied and ended up with a decent sketch of what Dr. Ouardaoui and the Office of International Programs (OIP) were essentially looking for. I had to confirm that personally and make sure that no other requirement was to be added, modified or deleted from the list: a meeting with Ms. Fishburn, the head of the OIP, was going to make all the needs crystal clear. I was asked by Ms. Fishburn to build a similar system to the one of Duke University in which students were able to search for any course offered by a partner institution and bring up its equivalent course locally. This website fulfilled a great part of the core requirements that were pinned down earlier during my first meeting with my supervisors giving me a concrete illustration of what my application was to look like.

Moving on to the more technical aspect of the construction of the application, I had to meet Ms. Lamiae Bouanane along with Dr. Naeem Sheikh in order to discuss the tools that were preferable for the making of the web application in order to maximize the compatibility for later
integration with the portal of the university. Ms. Bouanane, being part of the information systems sector of the Information Technology Services (ITS), is the system administrator and knows every detail about the services that are integrated to the portal. I was very happy to find out that AUI’s portal was based off ASP.NET and C# which I had already worked and gained familiarity with previously on multiple occasions, including the project of my internship.

The software engineering model that I chose to employ is the incremental model in which the client and the developer agree on a set of requirements and needs to be met in the final product and where the developer shows his or her progress to the client in a progressive manner, implementing the end result feature by feature, making sure that the client is satisfied.
REQUIREMENT GATHERING

We are to divide this section into two parts displaying the functional as well as the non-functional requirements of the management system.

Functional requirements:

- Two types of users/accounts:
  - Student: account for students in AUI
  - Administrator: account for the OIP and Dr. Ouardaoui

- The Student shall be able to:
  - Search for a course offered abroad by
    - Institution
    - Equivalent course at AUI

- The Administrator shall be able to:
  - Search for a course offered abroad by
    - Institution
    - Equivalent course at AUI
  - Add a course to the database with the following details:
    - Professor who approved the course
    - Date of approval of the course
  - Modify a course from the database
  - Delete a course from the database
Non-functional requirements:

- Performance: the web application shall be fast and responsive with an efficient use of RAM and bug-free
- Scalability: the database, even with more than 2,000 courses, shall not destabilize or dramatically slow down the processing speed of the system
- Compatibility: because this is a web application, it can be opened from any platform using a web browser such as Mozilla Firefox, Google Chrome or Internet Explorer
- Usability: the student shall be able to search for courses by very few clicks and the administrator, in addition to that, shall be able to insert a new course, modify or delete an already existing one because of the constant changes made by universities all over the world
TECHNOLOGY ENABLERS

A set of multiple tools were used to work on the project which are the following:

I have worked with MySQL Workbench multiple times before considering this capstone project. It is the DBMS (Database Management System) that I have used to design and implement the database of the web application. Because it is open-source and widely used by the internet community, obstacles encountered during the development were easy to tackle and get rid of.

Microsoft Visual Studio is my IDE (Integrated Development Environment) of choice when it comes to web application development as it is a powerful and complete tool with a wide range of components and libraries to suit the needs of the end result of the development process. It employs ASP.NET for the web design part and either C# or Visual Basic for the business part of the project.
ASP.NET is a framework, deployed by Microsoft, with the intent of making the development of web applications or dynamic web pages easier. It combines forces with languages such as HTML, CSS and JavaScript in order to deliver fully fleshed web solutions.

The main programming language that I have used for responding to the requirements is C# (also known as C ‘Sharp’). It is similar to Java and has an incredible amount of libraries that respond to many demands.

Last, but definitely not least, SQLizer is a simple online tool that saved me ages of work: it is able to convert a Microsoft Excel sheet into a series of SQL statements in order to fill the database with all of the data in a matter of very few clicks.
DESIGN AND IMPLEMENTATION

Model

Because of the time constraints, I decided to use the Incremental model of software engineering in order to maximize efficiency and client satisfaction. It is essentially a set of small Waterfall sets of requirements that are satisfied chunk by chunk and presented to the client in order to ensure that it is what the developer is asked to do.

Figure 1: adopted software engineering model

For instance in our case, Build 1 was the phase where the content of the database was fully linked to the web application and was displayed in the form of a table to the user of the web application. Upon completion of that, I met with my supervisors in order to present them what was done so far at that point and whether they were satisfied with the current progress or not. Moving on to the next build, I had the editing, deleting and adding of courses up and functional as well as the ability to search through the courses by host institution or AUI equivalent course code. The third and final build supports a login system that makes sure to give the correct
privileges mentioned in the requirement specifications, to the student and the administrator accordingly.

**Entity Relationship Diagram**

The following diagram shows the linkage that is happening between the four entities that we have stored in our database:

![Entity Relationship Diagram](image)

Figure 2: ERD used for the database
We can see that the database is centered on the course that is offered at a host institution: a faculty member can approve one or more courses, a course is approved by only one faculty member; an institution can offer one or more courses, yet a specific course is offered in one institution; a course offered at AUI can have one or more equivalent courses offered abroad, but a course offered on exchange has only one AUI course that is equivalent to it.

**Business Logic**

A set of functions was implemented to make the tasks of Dr. Ouardaoui and the OIP more automated and user-friendly and they are listed below:

- Displaying the courses: this function calls an SQL query which displays the list of courses offered abroad accordingly.

```sql
SelectCommand="SELECT *
FROM crs_at_host INNER JOIN institution ON
  crs_at_host.Nb_inst=institution.Nb_inst INNER JOIN
  crs_at_aui ON crs_at_host.Nb_crs_aui=crs_at_aui.Nb_crs_aui
WHERE institution.Inst_name like @Institution
AND crs_at_aui.Crs_code_aui like @Course_Code_at_AUI
ORDER BY Inst_name"
```

Figure 3: SQL statement behind the selection of the courses

- Deleting a course: this function is in charge of the deletion of a selected course from the database and can be used only by the administrator of the system.

```sql
DeleteCommand="DELETE FROM crs_at_host WHERE Nb_crs_host=@Nb_crs_host"
```

Figure 4: SQL statement behind the deletion of courses
• Modifying a course: this function gives the administrator the ability to modify the attributes of a certain course in the system.

```
UpdateCommand="START TRANSACTION;
UPDATE institution, crs_at_host, crs_at_aui
SET institution.Inst_name = @Inst_name,
crs_at_host.Crs_code_host = @Crs_code_host,
crs_at_host.Crs_title_host = @Crs_title_host,
crs_at_aui.Crs_credits_aui = @Crs_credits_aui,
crs_at_aui.Crs_code_aui = @Crs_code_aui,
crs_at_aui.Crs_title_aui = @Crs_title_aui
WHERE institution.Nb_Inst = crs_at_host.Nb_Inst
AND crs_at_aui.Nb_crs_aui = crs_at_host.Nb_crs_aui
AND crs_at_host.Nb_crs_host = @Nb_crs_host;
COMMIT;"
```

Figure 5: SQL statement behind the updating of the courses

• Adding a course: this function gives the ability to the administrator to add a new course into the database by inserting the name of the institution, the course code at that host institution, the course title, the number of credit hours acquired in AUI, the course code at AUI, the course title at AUI, the name of the professor who approved the course as well as the date of approval for statistics gathering.

```
START TRANSACTION;
INSERT INTO institution(Inst_name)
VALUES('"+TextBox1.Text+"');
INSERT INTO crs_at_aui(Crs_credits_aui, Crs_code_aui, Crs_title_aui)
VALUES('"+TextBox4.Text+", "+TextBox5.Text+", "+TextBox6.Text+"');
INSERT INTO faculty(Fac_name, Fac_dateapprov)
VALUES('"+TextBox7.Text+", "+TextBox8.Text+"');
INSERT INTO crs_at_host(Crs_code_host, Crs_title_host, Nb_inst, Nb_crs_aui, Nb_faculty)
VALUES('"+TextBox2.Text+", "+TextBox3.Text+", 
(SELECT Nb_inst FROM institution WHERE institution.Inst_name = 
"+TextBox1.Text+"),
(SELECT Nb_crs_aui FROM crs_at_aui WHERE crs_at_aui.Crs_code_aui = 
"+TextBox5.Text+"),
(SELECT Nb_faculty FROM faculty WHERE faculty.Fac_name = 
"+TextBox7.Text+"));
COMMIT;
```

Figure 6: SQL statement behind the addition of the courses
Interface

The result, giving us an overall look of the interface from the administrator’s perspective, is as follows:

![Interface View](image)

Figure 7: view of the interface from the administrator’s perspective
Below are screenshots of the drop down lists used as a searching tool in the application:

- Any -
- Any -

Abu Dhabi University
Ajou University South Korea
Akita International University
Alfred University
Allegheny College
American University in Kosovo
American University of Paris
American University of Science and Technology, beirut
Anglo American University of Prague
Appalachian State University
Arkansas State University
Baldwin-Wallace University
Baylor University
Bellarmine University
Beloit University
Benedict College
Bentley University
Binghamton University

Figure 8: combo box showing the list of universities in the database
Figure 9: combo box showing the codes of the courses in AUI
The user of the application may filter the results depending on what he or she selected in the drop down lists, allowing for a more narrow and accurate set of results. In this illustrative example, we supposed that the user wishes to look up the courses and their equivalences that are offered in Arkansas State University:

![Figure 10: zoom-in on the grid of courses](image)

<table>
<thead>
<tr>
<th>Institution</th>
<th>Course Code at Host</th>
<th>Course at Host Institution</th>
<th>Credit Hours at AUI</th>
<th>Course Code at AUI</th>
<th>Course at AUI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edit Delete Arkansas State University</td>
<td>MGMT 3183</td>
<td>Entrepreneurship</td>
<td>3</td>
<td>MGT 3302</td>
<td>Entrepreneurship</td>
</tr>
<tr>
<td>Edit Delete Arkansas State University</td>
<td>SOC 2213</td>
<td>Principles of Sociology</td>
<td>3</td>
<td>SOC 1301</td>
<td>Introduction to Sociology</td>
</tr>
<tr>
<td>Edit Delete Arkansas State University</td>
<td>ECON 3323</td>
<td>Money and Banking</td>
<td>3</td>
<td>FIN 3302</td>
<td>Money and Banking</td>
</tr>
<tr>
<td>Edit Delete Arkansas State University</td>
<td>MKTG 3033</td>
<td>Advertising and Promotion</td>
<td>3</td>
<td>MKT 3302</td>
<td>Advertising and Promotion Management</td>
</tr>
<tr>
<td>Edit Delete Arkansas State University</td>
<td>MKTG 4223</td>
<td>Marketing Management</td>
<td>3</td>
<td>MKT 4305</td>
<td>Marketing Management</td>
</tr>
</tbody>
</table>

Figure 10: zoom-in on the grid of courses

But then the user wishes to look up all of the courses that are offered in the latter university, but that are equivalent to the SOC 1301 course offered in AUI:
<table>
<thead>
<tr>
<th>Institution</th>
<th>Course Code at Host</th>
<th>Course at Host Institution</th>
<th>Credit Hours at AUI</th>
<th>Course Code at AUI</th>
<th>Course at AUI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arkansas State University</td>
<td>SOC 2213</td>
<td>Principles of Sociology</td>
<td>3</td>
<td>SOC 1301</td>
<td>Introduction to Sociology</td>
</tr>
</tbody>
</table>

Figure 11: zoom-in on the results based on the user’s selection
This user changed his or her mind about Arkansas State University but still needs to take the Sociology course during his or her upcoming semester abroad. He or she will look up all of the courses that are equivalent to AUI’s SOC 1301, but this time anywhere and not only at Arkansas State University:

Figure 12: zoom-in on the results depending on the user’s new selection
Our friend here is getting lazy and doesn’t wish to go all the way up in the course code list to get back to the full grid of courses, well worry no more! The ‘Reset’ button has a very self-explanatory title to bring back up all of the courses offered abroad with the click of a button:

![Figure 13: zoom-in on the table after clicking on the ‘Reset’ button](image)

<table>
<thead>
<tr>
<th>Institution at Host</th>
<th>Course Code at Host</th>
<th>Course at Host Institution</th>
<th>Credit Hours at AUI</th>
<th>Course Code at AUI</th>
<th>Course at AUI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Abu Dhabi University</td>
<td>MMC 201</td>
<td>Introduction to Mass Communication</td>
<td>3</td>
<td>COM 1302</td>
<td>Introduction to Mass Communication</td>
</tr>
<tr>
<td>Abu Dhabi University</td>
<td>ECO 201</td>
<td>Microeconomics Analysis and Application</td>
<td>3</td>
<td>ECO 2301</td>
<td>Microeconomics</td>
</tr>
<tr>
<td>Appalachian State University</td>
<td>TEC 1708</td>
<td>Construction Technology &amp; Building Codes</td>
<td>3</td>
<td>__</td>
<td>Thematic Area</td>
</tr>
<tr>
<td>Appalachian State University</td>
<td>TEC 2758</td>
<td>Surveying, Soils &amp; Foundations</td>
<td>3</td>
<td>__</td>
<td>Thematic Area</td>
</tr>
<tr>
<td>Appalachian State University</td>
<td>TEC 2718</td>
<td>Building Mechanical Systems</td>
<td>3</td>
<td>__</td>
<td>Thematic Area</td>
</tr>
</tbody>
</table>

Figure 13: zoom-in on the table after clicking on the ‘Reset’ button
If the user wishes to edit the content displayed in a row, simply clicking the ‘Edit’ button for the course of his or her choice will allow modifications to be made freely:

<table>
<thead>
<tr>
<th>Nb</th>
<th>Institution</th>
<th>Course Code at Host</th>
<th>Course at Host Institution</th>
<th>Credit Hours at AUI</th>
<th>Course Code at AUI</th>
<th>Course at AUI</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Abu Dhabi University</td>
<td>MMC 201</td>
<td>Introduction to Mass Comm</td>
<td>3</td>
<td>COM 1302</td>
<td>Introduction to Mass Comm</td>
</tr>
<tr>
<td>2</td>
<td>Abu Dhabi University</td>
<td>ECO 201</td>
<td>Microeconomics Analysis</td>
<td>3</td>
<td>ECO 2301</td>
<td>Microeconomics</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>and Application</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>Ajou University South Korea</td>
<td>1035</td>
<td>Financial Management</td>
<td>3</td>
<td>FIN 3303</td>
<td>Corporate Financial Management</td>
</tr>
</tbody>
</table>

Figure 14: zoom-in on the row being edited
Adding a course to the database can be done by simply filling in the fields accordingly:

**Add a course:**

- Institution
- Course Code at Host
- Course title at Host Insitution
- Credit Hours at AUI
- Course Code at AUI
- Course Title at AUI
- Approved by
- Date (YYYY-MM-DD)

Figure 15: fields that get filled to add a course
All of these functionalities are achieved by administrators of the application. The student has only access to the list of courses and the searching capabilities:

![Figure 16: view of the interface from the student’s perspective](image-url)
CONCLUSION

This capstone project has helped me strengthen my coding skills and get an even deeper understanding of MySQL and Microsoft Visual Studio and how they work, especially with all of the new libraries that I discovered were available for Visual Studio. Every single obstacle actually enhanced the learning experience by bringing in more variety and more questions to be answered, therefore making me a wiser programmer, which brings us to the challenges that were faced throughout the semester.

Because I did not take capstone design alone but with other courses as well, my focus shifted regularly between the project and classes. I was not able to give my full attention to the web application during the entirety of the semester. I had trouble meeting Ms. Fishburn earlier during the semester as she was not in her office for a certain period of time. Not to mention all of the unforgettable bugs and unexpected behaviours of the IDE when testing the application, even with the code remaining unmodified from the last stable test. On top of all, the main challenge was the time interval that I had to complete the work but like I said previously, that pressure helped me improve my capacities of problem-solving under pressure.

This web application has a lot to improve upon, it is therefore important to note down the future work that could potentially be achieved through the timeline of the project. A more sophisticated way of gathering statistics should be beneficial to the administrators of the application in order to backtrack any source of any issue should it arise. More details about the courses should be added for an even more accurate search such as the country of the institution or the department of the course.
All in all, working on a computer science-related project always comes with its bugs and annoying errors that teleport out of nowhere and make your life harder, but what a feeling it is to fix all of them and get a fully working product at your hands! It is those red lines and crashes and, to be honest, the insane amount of procrastination that made me a better engineer and increased my problem-solving capabilities within a limited timeframe.
**WEBSITES**

http://stackoverflow.com/
https://sqlizer.io

**APPENDICES**

Appendix A: ASP.NET code behind the display of the table

```xml
<asp:GridView ID="GridView1" runat="server" DataSourceID="SqlDataSource1" AutoGenerateColumns="False" style="z-index: 1; width: 989px; height: 238px; position: absolute; top: 222px; left: 445px; AllowPaging="True" AutoGenerateDeleteButton="True" AutoGenerateEditButton="True" AutoGenerateInsertButton="True" AutoGenerateFilterButton="True" AutoGenerateOrderByButton="True" AutoGenerateSortButton="True" AutoGenerateViewButton="True" DataKeyNames="Nb_cr_s_host" CellPadding="4" EnableTheming="True" ForeColor="#333333" GridLines="None">
   <AlternatingRowStyle BackColor="#000000"/>
   <Columns>
      <asp:BoundField DataField="Nb_cr_s_host" HeaderText="Nb" SortExpression="Nb_cr_s_host" InsertVisible="False" ReadOnly="True"/>
      <asp:BoundField DataField="Inst_name" HeaderText="Institution" SortExpression="Inst_name"/>
      <asp:BoundField DataField="Cr_s_code_host" HeaderText="Course Code at Host" SortExpression="Cr_s_code_host"/>
      <asp:BoundField DataField="Cr_s_title_host" HeaderText="Course at Host Institution" SortExpression="Cr_s_title_host"/>
      <asp:BoundField DataField="Cr_s_credits_aui" HeaderText="Credit Hours at AUI" SortExpression="Cr_s_credits_aui"/>
      <asp:BoundField DataField="Cr_s_code_aui" HeaderText="Course Code at AUI" SortExpression="Cr_s_code_aui"/>
      <asp:BoundField DataField="Cr_s_title_aui" HeaderText="Course at AUI" SortExpression="Cr_s_title_aui"/>
   </Columns>
</asp:GridView>
```
Appendix B: code behind the mechanics of the drop down lists

```xml
<SelectParameters>
  <asp:ControlParameter ControlID="DropDownList1" Name="Institution" PropertyName="SelectedValue"/>
  <asp:ControlParameter ControlID="DropDownList2" Name="Course_Code_at_AUI" PropertyName="SelectedValue"/>
</SelectParameters>
```
Appendix D: choice of colors to mimic AUI’s portal

```xml
<EditRowStyle BackColor="#7C6F57" />
<FooterStyle
    BackColor="#02631C"
    Font-Bold="True"
    ForeColor="White" />
<HeaderStyle
    BackColor="#02631C"
    Font-Bold="True"
    ForeColor="White" />
<PagerSettings
    PageButtonCount="7"
    Mode="NumericFirstLast" />
<PagerStyle
    BackColor="White"
    ForeColor="#006666"
    HorizontalAlign="Center"
    BorderStyle="None" />
<RowStyle BackColor="#E3EAEB" />
<SelectedRowStyle
    BackColor="#C5BBAF"
    Font-Bold="True"
    ForeColor="#333333" />
<SortedAscendingCellStyle BackColor="#F8FAFA" />
<SortedAscendingHeaderStyle BackColor="#246B61" />
<SortedDescendingCellStyle BackColor="#D4DFE1" />
<SortedDescendingHeaderStyle BackColor="#15524A" />
</asp:GridView>
```