**Bachelor’s Degree Report (BSc)**

**Course Catalogue Handling System (CCHS) A Project Report**

**BY**

**Mohammed Mustapha AHMED (A00016681)**

**Supervisor: Alexey Vedischev**

**School of Information, Technology and Computing American University of Nigeria**

**May 2017**

**Bachelor’s Degree Report (BSc)**

**Course Catalogue Handling System (CCHS) A Project Report**

**BY**

**Mohammed Mustapha AHMED (A00016681)**

**Supervisor: Alexey Vedischev**

**School of Information, Technology and Computing American University of Nigeria**

**May 2017**

**DECLARATION**

I declare that this project work was carried out by Mustapha has not been previously submitted for the degree. And that the report is written unaided in my own words, apart from any quoted material, which I identified clearly in the correct manner and fully acknowledged work by others. The work and the report were carried out under the guidance of Alexey Vedischev.

Student's Name:

Mohammed Mustapha Ahmed Signature....................................

Supervisor's Name:

Alexey Vedischev Signature.....................................

Program Chair's Name:

Dr. Narasimha Rao Vajjhala Signature....................................

Dean's Name:

Dr. Mathias Fonkam Signature.....................................

SDP Coordinator’s Name:

Dr. Abubakar S. Hussaini Signature.....................................

# ABSTRACT

In an academic setting, it is important for students to be well informed about their academic status and progress regularly. All first year students are provided with school catalogs which hold information about their majors and the courses required by them to pass for them to obtain their particular degrees. Students are given these catalogs only once and many of them tend to misplace them after a few semesters. Hence, they lose information on the requirements they need to obtain a degree in the university within the promised four year period. This causes many students to lose track of the time they are supposed to complete all their requirements. When students do not obtain their degrees within the promised 4 year period, questions start being asked by students and their parents as well. With a relatively expensive tuition fee, it could be a major problem for parents to pay for tuition for extra semesters which they had not budgeted for. The department responsible for keeping track of the academic progress of students is the Academic Advising Office. This office is not doing particularly badly at their duties but a lot more can be done. Firstly, the location of the office is constantly being changed almost every semester. Secondly, students who do not have good academic standings feel discouraged or embarrassed to visit the Academic Advising and reveal information about their grades and cumulative grade point average (C|GPA). Thirdly, many students might not remember all the grades they have gotten from the courses they have taken especially when they are in the latter stages of their degree. The school provides check sheets for students but many do not feel the need to obtain them, the distance from the dorms to the registrar’s office might be a discouraging factor. In addition, even when students obtain these check sheets, these sheets get misplaced for

one reason or the other. This would mean all the information about their grades and C|GPA become lost.

The Catalog Handling system (CHS) would make all school year catalogs available to students, the registrar, deans’ offices and the academic advising office at all times given that internet connection is made available. Every student will be required to insert their courses grades at the end of every semester. Students will be able to insert their course grades and calculate their C|GPA every semester and the academic advising office will have access to every student records. These student records will include statistical report of every student’s academic progress over each semester. This is the main motive of the Catalog Handling System (CHS). The deployment of this system will help students become more familiar with their academic requirements and also aid the academic advising office in keeping them on track in order to obtain their degree timely and successfully.

# DEDICATION

I, Mohammed Mustapha Ahmed, dedicate this work to my father, Alhaji A.O Ahmed and my mom, Mrs. Amina Ahmed who have been instrumental throughout my education. They have always inspired me to push myself further academically and work harder in life as a whole.

# ACKNOWLEDGEMENT

I would like to give all praise and gratitude to my supervisor, Alexey Vedischev, for his support and guidance throughout the development of this project. His patience and perseverance with me throughout the development of this project made it easier for me to express my thoughts and opinions. I would also like to give thanks to professor Evgeny Arkhipov for his assistance whenever I had hit a road block during the development of this project. Finally, I would like to give special thanks to my family and friends who have offered me support, words of encouragement and love throughout my education.

Table of Contents

[DECLARATION i](#_bookmark0)

[ABSTRACT ii](#_bookmark1)

[DEDICATION iv](#_bookmark2)

[ACKNOWLEDGEMENT v](#_bookmark3)

[LIST OF FIGURES viii](#_bookmark4)

[LIST OF TABLES ix](#_bookmark5)

[CHAPTER ONE 10](#_bookmark6)

1. [INTRODUCTION 10](#_bookmark7)
	1. [BACKGROUND 10](#_bookmark8)
	2. [PROJECT AIMS 10](#_bookmark9)
	3. [PROJECT OBJECTIVE 11](#_bookmark10)
	4. [OUTLINE OF THE REPORT 11](#_bookmark11)

[CHAPTER TWO 12](#_bookmark12)

1. [REQUIREMENTS 12](#_bookmark13)
	1. [FUNCTIONAL REQUIREMENTS 12](#_bookmark14)
	2. [NON-FUNCTIONAL REQUIREMENT 13](#_bookmark15)
	3. [USECASE DIAGRAM 14](#_bookmark16)
	4. [USE CASE DESCRIPTION 14](#_bookmark17)
	5. [STATE DIAGRAM 22](#_bookmark19)
	6. [SYSTEM ARCHITECTURE 26](#_bookmark20)
	7. [ENTITY RELATIONSHIP DIAGRAM 26](#_bookmark21)
	8. [HARDWARE REQUIREMENT 27](#_bookmark22)

[CHAPTER THREE 28](#_bookmark23)

1. [DESIGN 28](#_bookmark24)
	1. [METHODOLOGY 28](#_bookmark25)
	2. [ARCHITECTURAL DESIGN 29](#_bookmark26)
		1. [CLIENT TIER 29](#_bookmark27)
		2. [MIDDLE-TIER 29](#_bookmark28)
		3. [DATABASE TIER 29](#_bookmark29)
	3. [TECHNOLOGIES USED 30](#_bookmark30)
		1. [HARDWARE 30](#_bookmark31)
		2. [SOFTWARE 30](#_bookmark32)
		3. [LANGUAGES 30](#_bookmark33)

[CHAPTER FOUR 31](#_bookmark34)

1. [CONCLUSION 31](#_bookmark35)
	1. [RECOMMENDATION 31](#_bookmark36)
	2. [CONCLUSION 31](#_bookmark37)
	3. [FUTURE WORK 32](#_bookmark38)

[REFERENCES 33](#_bookmark39)

# LIST OF FIGURES

*Figure 1: Use case diagram Figure 2: State Diagram (Admin)*

*Figure 2.1: State Diagram (Student)*

*Figure 2.2: State Diagram (Academic Advising) Figure 2.3: State Diagram (Instructor)*

*Figure 3: System Architecture*

*Figure 4: Agile Software Development Cycle*

# LIST OF TABLES

[*Table 1: Use Cases* 1](#_bookmark18)7

# CHAPTER ONE

1. **INTRODUCTION**

## BACKGROUND

This project is a web application that will make all school catalogues available to all AUN student and faculty at all times. It will also serve as web application where students can receive their grades from instructors and know their academic standing by calculating their C|GPAs. Another primary aim of the project is to enable the academic advising office keep track of all students’ records. The Admin (Register) will be logged into the system. He/She will be able to create and modify a catalog. Students, Faculty and Academic advising officers will be logged in the system to view all school catalogues. Students will be logged in and will be able to calculate their C|GPA. Instructors will be logged in and will be able to submit grades to the Admin (Registrar). Academic advising officers will be logged into the system and will be able to view all students’ grades. The system can be accessed at all times in all platforms as long as there is internet connection.

## PROJECT AIMS

The development of this project would make school catalogues easily accessible to students and faculty at all times. It will also make it easier for the academic advising

office to keep track of students’ academic progress. Students will be able to calculate

their C|GPA. Instructors will be able to submit students’ grades in order for students to know their academic standing after each semester.

## PROJECT OBJECTIVE

The proposed Web-based application has the following objectives:

* + - To enable faculty to submit grades.
		- To enable AUN faculty and students to access all school catalogues.
		- To enable academic advising keep track of students’ records.
		- To enable students to calculate their C|GPA.

## OUTLINE OF THE REPORT

**Chapter 1: Introduction**

* The aims and objectives of the development of the project

**Chapter 2**: **Requirements**

* The functional and non-functional requirements of the project

**Chapter 3**: **Design**

* The architecture and design of the project

**Chapter 4**: **Conclusion**

* The conclusion of the project

**CHAPTER TWO**

1. **REQUIREMENTS**

Requirements are constraints that describe how the system software should function as well as what the system should do or have. In this phase of the project, the functional requirements (which is the behavior of the system) and non-functional requirements (which are the system attributes) will be identified. It also consists of the hardware requirements that a user needs to access the system.

## FUNCTIONAL REQUIREMENTS

The functional requirements state the behavior of the system software. The functional requirements are:

* The system would allow the Admin to create catalogues
* The system would allow the Admin to view catalogues
* The system would allow the Admin to manipulate catalogues
* The system would allow the Admin to view student records
* The system would allow instructors to submit grades
* The system would allow instructors to view catalogues
* The system would allow academic advising to view catalogues
* The system would allow Admin to manage user details
* The system would allow students to view catalogues
* The system would allow students to calculate cumulative grade point average(CGPA)

## NON-FUNCTIONAL REQUIREMENT

The non-functional requirements are:

**Look and Feel Requirements -**The intended outlook of the software will appeal to the users and its appearance will be appropriate for an academic institution.

**Usability Requirements -**The system requires the users to be computer literates. Nevertheless, the product will be easy to use.

**Performance Requirements -** School catalogues will be easily accessible for all users, student progress will be monitored efficiently and students’ lifestyles will be enhanced academically.

**Operational Requirements** – The system will be used by students, the academic advising office and the registrar of the American University of Nigeria.

**Maintainability and Portability Requirements** – The system will initially be made available as applications and it will eventually be made available for ios and android users.

**Security Requirements** – User details will be restricted and secured.

**Cultural and Political Requirements** – N/A

**Legal Requirements –** The system will adhere to all laws, rules and regulations regarding data restriction

## USECASE DIAGRAM

The use case diagram below shows the functional requirements of the system and how each user initiates a functionality. There will be four (4) users of the system. The users are the Registrar (Admin), Student, Instructor and Academic Advisor.

\

***Figure 1: Use case diagram***

## USE CASE DESCRIPTION

#### Table 1: Use Cases

|  |  |
| --- | --- |
| 1. | Login |
| 2. | Create Catalog |
| 3. | Modify Catalog |
| 4. | Delete Catalog |
| 5. | View Catalogues |

|  |  |
| --- | --- |
| 6. | View User Records |
| 7. | Add Course |
| 8. | Add Department |
| 9. | Calculate C|GPA |
| 10. | Submit Grades |
| 11. | Insert Grades |
| 12. | Add Academic Advising |

**Use-case Documentation**

**Identifier and Name:** UC1 Login

**Initiator:** Student, Academic Advising, Admin

**Goal:** To make access to the system

**Pre-condition:** The User is a registered student or staff of the American University of Nigeria

**Post-condition:** Actor is registered as a user

**Assumptions:** User is a valid registered member of the university

**Main success scenario:**

1. User must obtain login details via registration
2. User logs on website
3. Website prompts user for registration details
4. User inserts details
5. User gains access to the website

**Identifier and Name:** UC2 Register

**Initiator:** Student

**Goal:** To register students in the system with their respective login details

**Pre-condition:** Student has a school identification number

**Post-condition:** Student is registered and obtains login details **Assumptions:** User is a valid registered member of the university **Main success scenario:**

1. User accesses website
2. Website prompts user for registration details
3. User inserts details
4. User is registered on the system

**Identifier and Name:** UC3 Create Catalog

**Initiator:** Admin

**Goal:** Create a student catalog

**Pre-condition:** Catalog must be approved by the deans and chairs

**Post-condition:** Catalog is added to the database **Assumptions:** All departments are included in the Catalog **Main success scenario:**

* 1. Catalogue is created
	2. Departments are created in the Catalog
	3. Courses in the departments are created and listed in their appropriate sections

**Identifier and Name:** UC4 View Catalog

**Initiator:** Student, Admin, Academic Advising

**Goal:** To gain access to a school catalog

**Pre-condition:** User is a registered member of the university

**Post-condition:** Catalog is viewed

**Main success scenario:**

1. User selects the option to view Catalogs
2. All Catalogs are displayed

**Identifier and Name:** UC5 Modify Catalog

**Initiator:** Admin

**Goal:** Admin modifies Catalog

**Pre-condition:** Catalogue is available **Post-condition:** Modification is successful **Main success scenario:**

1. Admin selects the “Edit Catalog” option
2. Admin updates old details with new details
3. Admin saves changes

**Identifier and Name:** UC6 Delete Catalog

**Initiator:** Admin

**Goal:** Admin deletes Catalog

**Pre-condition:** Catalog is available **Post-condition:** Deletion is successful **Main success scenario:**

1. Admin selects the “Delete Catalog” option
2. Catalog is deleted
3. Admin saves changes

**Identifier and Name:** UC7 View all student records

**Initiator:** Admin, Academic Advising

**Goal:** Restricted users will be able to view all student records

**Pre-condition:** All students are registered and making use of the system

**Post-condition:** All student records are available to be viewed

**Main success scenario:**

1. Restricted user selects the “View Student Records” option
2. All student records will be displayed

**Identifier and Name:** UC8 Insert Grades

**Initiator:** Student

**Goal:** Students inserts grades according to their respective courses

**Pre-condition:** All students have gotten their grades for their respective courses

**Post-condition:** Cumulative grade point average(CGPA) is viewed

**Main success scenario:**

* 1. Student selects “Check sheet” option
	2. Department is selected
	3. Major is selected
	4. Course is selected
	5. Grades are inserted

**Identifier and Name:** UC9 Add Academic Advising

**Initiator:** Admin

**Goal:** To add a registered academic advising staff to the system

**Pre-condition:** Academic advising staff is a registered member of the university

**Post-condition:** Academic advising staff is added

**Main success scenario:**

1. Admin adds an Academic Advising staff to the database
2. Admin saves the staff in the database

**Identifier and Name:** UC10 Add Department

**Initiator:** Admin

**Goal:** To add school departments into database

**Pre-condition:** Department is confirmed by the deans and chairs

**Post-condition:** Department is saved in the database

**Main success scenario:**

1. Admin adds department into the database
2. Admin saves the department

**Identifier and Name:** UC11 Add Course

**Initiator:** Admin

**Goal:** To add courses into database

**Pre-condition:** Course is confirmed by the deans and chairs

**Post-condition:** Course is saved in the database

**Main success scenario:**

1. Admin adds course into database
2. Admin saves the course

**Identifier and Name:** UC12 Submit Grades

**Initiator:** Instructor

**Goal:** To submit grades to registrar

**Pre-condition:** Course has been completed for the semester

**Post-condition:** Grades are submitted to registrar

**Main success scenario:**

1. Instructor clicks “Submit Grades”
2. Instructor inserts student grades
3. Instructor submits grades to registrar

**Identifier and Name:** UC13 View User Records

**Initiator:** Admin

**Goal:** To view and manipulate user records

**Pre-condition:** User must be registered into the system

**Post-condition:** User records have been viewed or modified

**Main success scenario:**

1. Admin clicks “View User Records”
2. Admin views or modifies user data
3. Admin saves user data

**Identifier and Name:** UC14 Calculate C|GPA

**Initiator:** Student

**Goal:** To calculate cumulative grade point average **Pre-condition:** User is registered into the system **Post-condition:** C|GPA has been calculated

**Main success scenario:**

1. Student clicks “Calculate C|GPA”
2. Student inserts grades
3. Student clicks “submit”
4. Calculator shows result

## STATE DIAGRAM

A state diagram describes the behaviour of an object. In other words, it is an illustration of the states an object can have as well as the changes or transitions that can occur within its lifetime. The following diagrams are the state diagrams for different actors of the system;



***Figure 2: State Diagram (Admin)***



***Figure 2.1: State Diagram (Student)***



***Figure 2.2: State Diagram (Academic Advising)***



***Figure 2.3: State Diagram (Instructor)***

## SYSTEM ARCHITECTURE

A system architecture is a model that identifies the physical view and components of the system and the relationships among them.



***Figure 3: System Architecture***

## ENTITY RELATIONSHIP DIAGRAM



## HARDWARE REQUIREMENT

To access the system, a user is required to have the following:

* + - A computer, a phone or a tablet
		- Internet connection

# CHAPTER THREE

# DESIGN

## METHODOLOGY

The software process model used in this project is called the Agile Methodology. This methodology. It consists of cycles. Each of these cycles consists of Requirements gathering, Architecture and Design, Development, and Test Feedback. This process is iterative i.e. Each completion of a phase is repetitive to make sure it corresponds to the other phases and also to ensure the software is of good quality.



***Figure 4: Agile Software Development Cycle***

## ARCHITECTURAL DESIGN

CHS consists of a three-tier architectural design. It involves the front-end (client-tier), the middleware (web server tier) and the back-end (database tier).

### CLIENT TIER

The front end is the part of the system where the users directly interact with the system’s functionalities. This is the part of the system that the users can see. The technologies used in this part of the development of the Catalogue Handling System (CHS) are HTML, PHP, Laravel, CSS and JavaScript. In this section, the user sends information to the web server and database server. In return, it displays the output of the information sent by the user.

### MIDDLE-TIER

Middle-tier is the part of the system that performs the system logic. This part integrates the user interface to the database using PHP 7, a version of PHP. PHP 7 receives data from users and sends the data to the database server. It can also receive data from the database server and send it to users.

### DATABASE TIER

Back-end tier is the part of the system that interacts with the front-end tire indirectly. The database server is the back-end tier for the system. Only the developers of the

system can have access to the system. MySQL is the database tier for the system.

## TECHNOLOGIES USED

### HARDWARE

* Computer

### SOFTWARE

**LARAVEL:** A PHP Framework

**MySQL**: The database server that was used/

**XAMPP**: A platform that consists of the local web server(Apache) and the database (MySQL)

**APACHE** : The local web server embedded in XAMPP

### LANGUAGES

**HTML:** It was used to design the web application for different web browsers

**CSS:** Acronym for Cascading Style Sheet. It was used in designing the client tier

**PHP:** The scripting language which was used to connect the client tier to the database tier

**JavaScript:** A scripting language used to make the front-end tier more interactive

# CHAPTER FOUR

# CONCLUSION

## RECOMMENDATION

* School catalogues should not be made with paper anymore.
* Students should be registered into the system right from enrollment.
	1. **CONCLUSION**

Catalogues hold vital information about course requirements needed in completing every degree program. Catalogues are very important but currently they are either easily misplaced or destroyed. The AUN community would greatly benefit from a web based attendance system that would make school catalogues available to all students and faculty. This web based system would make school catalogues easily accessible to the AUN community. It would aid the academic advising office in keeping track of students’ academic progress.

* 1. **FUTURE WORK**

In future works, I would suggest the following in order to improve CHS:

* Platform for communication between the Academic Advising office and students
* Platform for communication between students and faculty.

# REFERENCES

1. E.Dr.Cachia, UML Fundamentals, 2001-2004.
2. A. Dennis, W. H. Barbara and R. M. Roberta, System Analysis and Design, 5th ed., Indiana: John Wiley and Sons,

Inc., 2011.