A FINAL YEAR RESEARCH PROJECT

ON

THE DESIGN AND IMPLEMENTATION OF A PROGRESSIVE CLOUD BASED LEARNING MATERIAL MANAGEMENT APPLICATION

PRESENTED TO

THE DEPARTMENT OF COMPUTER SCIENCE

FACULTY OF NATURAL SCIENCES

UNIVERSITY OF JOS

BY

OGEZI MICHAEL OKIBE

(UJ/2014/NS/0219)

IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THE AWARD OF BACHELOR OF SCIENCE IN

COMPUTER SCIENCE

MAY, 2019

Table of Contents

List of Figures

List of Tables

Dedication

I would like to dedicate this body of work to my parents and five siblings. I deeply appreciate them for all the love and support that they have all shown to me over the years.

Acknowledgments

I would like to thank the members of the Department of Computer Science and by extension the University of Jos. The amazing experiences that I have had these last few years would not have been possible without them.

I would also like to appreciate my supervisor, Dr. Mary-Jane Sule for all the help and guidance during my work on my final year project.

I would also like to thank Dr. Gideon Dadik, my Level Coordinator, and Dr. Yinka

Oyerinde who have been extremely helpful figures during my time at the University.

I would also like to thank my dad, Prof. A. E. Ogezi for all the love and support he gave me towards my work.

This acknowledgment would not be complete without appreciating my classmates and other colleagues for all the ideas we have shared over the years.

Finally, I would like to thank God.

Definitions of Operational Terms

1. **Cloud Computing (CC)** - A computing paradigm which involves applications being run and data being stored majorly in remote server farms and accessed by the end users through thin clients such as a browser
2. **Cloud Application (CA)** - A software application that is run on cloud computing infrastructure
3. **Cloud Storage (CS)** - A service that lets you store data by transferring it over the Internet or another network to an off-site storage system maintained by a third party
4. **Private Cloud** - Services offered over the Internet or over a private internal network to only select users, not the general public
5. **Public Cloud -** Services offered over the public Internet and available to anyone who wants to purchase them.
6. **Server-less Computing -** A computing model in which the cloud provider provisions and manages servers. It enables developers to spend more time building apps and less time managing infrastructure
7. **Hybrid Cloud -** A cloud that combines public and private clouds, bound together by technology that allows data and applications to be shared between them. A hybrid cloud gives businesses greater flexibility to scale up and down and offers more deployment options.
8. **Elastic Computing -** The ability to dynamically provision and de-provision computer processing, memory and storage resources to meet changing demands without worrying about capacity planning and engineering for peak usage.
9. **Cloud Bursting -** A configuration which is set up between a private cloud and a public cloud. If 100 percent of the resource capacity in a private cloud is used, then overflow traffic is directed to the public cloud using cloud bursting
10. **SAAS** (Software as a Service) - An application delivered over the Internet by a provider. Also called a hosted application. The application does not have to be purchased, installed or run on users' computers. SAAS providers were previously referred to as ASPs (Application Service Providers).
11. **EAAS** - Education as a Service
12. **TAAS** - Testing as a Service
13. **Mobile Application**
14. **Mobile Learning (M-Learning)**
15. **E-Learning**
16. **CBT** - Computer Based Testing
17. **Amazon EC2** - Amazon Elastic Compute Cloud is a central part of Amazon's cloud computing platform: Amazon Web Services (AWS). EC2 allows users to rent virtual computers on which to run their own computer applications

18. **API (Application Programming Interface)** - is a software program that facilitates interaction with other software programs. AVD: An Android Virtual Device is an emulator
19. **iOS** - A mobile operating system developed and distributed by Apple for iPhones, iPads and some iPods.
20. **Android** - An open-source mobile operating system developed by Google Inc.
21. **IAAS (Infrastructure As A Service)**: Infrastructure as a Service is a service model that provides storage, infrastructure and other hardware assets as resources that clients can provision.
22. **PAAS (Platform As A Service)** - A computing platform (operating system and other services) delivered as a service over the Internet by a provider. An example is an application development environment that you can subscribe to and use immediately.
23. **3G -** short for third Generation, is the third generation of mobile telecommunications technology.
24. **4G -** short for fourth generation, is the fourth generation of mobile phone mobile communication technology standards.
25. **Moodle** - A open source school learning management system currently in use by the University of Jos.
26. **P2P (Peer to Peer) Network** - This is a group of two or more devices each of which acts as a node for sharing files within the network. Instead of having a central server acting as a shared drive, each computer acts as the server for the files stored upon it.
27. **Wi-Fi** - This is a family of radio technologies that is used for the wireless local area networking of devices (WLAN) of devices which is based around the IEEE 802.11 family of standards. It is commonly by wrongly thought to mean Wireless Fidelity.
28. **IEEE 802.11** - This is part of the IEEE 802 family of LAN protocols which specify the set of MAC and physical layer protocols used for implementing WLAN.
29. **Wi-Fi Hotspot** - This is a physical location through which devices can obtain network access, typically using Wi-Fi technology.
30. **QR (Quick Response) Code** - This is a machine readable code consisting of an array of black and white squares, typically used for storing URLs and other information readable by the camera on a smartphone.
31. **HTTP/HTTPS**
32. **Skeuomorphism** - This term is used to in graphical user interface design to describe interface objects that mimic their real world counterparts in how they appear or can be interacted with. It helps ease the transition from a physical to software based solution. An example of this transition is one from a pocket calculator to a calculator app.
33. **SWOT –** A tool used to highlight the Strengths, Weaknesses, Opportunities and Threats involved with a proposed system.
34. **SQL (Structure Query Language) –** A domain-specific database language designed for managing a manipulating data held in relational databases.

35. **NoSQL –** This describes any means of storing data that is not SQL.
36. **GZIP –** A lossless data compression format.

Abstract

Education is a key factor in ensuring economic growth, especially for countries with quickly growing economies. Today, students have become more technologically savvy as teaching and learning practices make use of more advanced technology. Due to benefits such as increased flexibility and reduced costs, cloud computing is being adopted by more and more organizations every day.

Today, there is increased pressure on educational institutions to deliver better services using fewer resources (whether financial or human). Learning institutions, both public and private can glean the numerous benefits of cloud computing in order to ensure high quality service regardless of the location of the learner, teacher or management.

Cloud computing is taking a center stage in academia because of its myriad benefits. Various learning institutions use different cloud-based applications such as Google Docs, Office 365 and Quizlet to ensure that their students and other users can perform both academic and business-related tasks effectively. The software solutions based by cloud technology ensure that the research and development, as well as teaching and learning, are more sustainable and efficient, thus positively influencing the quality of our educational institutions. The myriad benefits of cloud technology have led to learning institutions across the globe adopting it as a solution to the various technological they face on a daily basis.

This project will seek to establish the benefits associated with the use of cloud computing in learning institutions, as well as propose a software solution tailored to the Nigerian context that will afford students even better learning experiences.

Chapter One: Introduction

1.1 Preamble

This chapter will introduce this research work by discussing the background of study, problem statement, objectives as well as the scope and limitation of the research.

Also contained in this chapter is the aim of the research, the research objectives and the research methodology that will be employed to achieved the said objectives.

1.2 Background to the study

The advent of cloud and mobile based solutions have enabled many useful advances in the lives of human beings. Cloud technologies continue to be implemented in mission critical and data intensive sectors such as energy (Sule, 2015).

I became fascinated with this field of study even before I stared pondering a research topic. This is because I felt that studying it could prove beneficial to the multitudes of students currently passing through our educational system. The current educational system and by extension the larger society has many problems; from poor job prospects for recent graduates to debilitating educational facilities and infrastructure. A paper that would examine our educational problems extensively and do them justice is a topic for another time. For now, I shall be looking at how cloud technology can be used to stem a few of these problems which ample studies have shown can be attenuated by intelligent applications of cloud technology and other technologies ranging from Mobile Apps and Artificial Intelligence (AI) to Virtual Reality (VR) and Augmented Reality (AR).

Mobile and Cloud technologies allow students and teachers to perform their work from anywhere. This is important for when these parties are outside the physical bounds of the learning institution. Removing the significance of physical location in the learning process will provide a boon to all stakeholders. These applications can allow students to practice for exams and tests while on the go; outside of classrooms and libraries.

AI can help students and teachers find more effective learning and teaching practices. It may even help schools to synthesize learning programs that are tailored to individual students. VR and AR technologies allow schools to teach their students in more engaging and memorable ways. These technologies have shown and continue to show great promise, but for the scope of this project, I will be focusing on mobile and cloud technology and what they bring to the field of education in Nigeria.

I will subsequently talk about some of the problems with our current educational system and propose an empirical approach to solving them.

1.3 Problems with the Current Educational System

The current education system is not perfect as common knowledge says. Problems with the current educational system are as follows:

1. Students find it difficult to access course material from anywhere.
2. Teachers are not always able to pass on new learning materials to their students in time.
3. The school administration may find it difficult to analyze data about their students across multiple software analytics tools. E.g. It is currently impossible to export data from Moodle to Big Data and data mining tools.
4. Students may find it difficult to practice for upcoming tests and exams when the course material is large and draws from multiple discrete sources.

1.4 Problem Statement

There is no cloud and mobile based learning material management system available to the students of the University of Jos.

There is Moodle; but it is neither cloud based nor optimized for mobile devices.

1.5 Objectives of the Study

The primal objective of this study is to unearth solutions that will improve the overall learning process for students in Nigeria using mobile and cloud technologies. These technologies when applied in other fields of endeavor have produced substantial forward strides for those involved. More specifically, I will be looking for ways in which mobile and cloud solutions can help students learn and prepare for exams better and help lecturers and teachers to teach and grade better.

The primary objective of the study is;

- To design and implement a system that will allow students learn more easily and efficiently using mobile and cloud technologies.

To this end the three sub-objectives that shall guide the implementation of the system have been outlined. They are;

- To develop a system that facilitates students' easy and effective access to learning material.
- To identify the most critical features of such a system as well as existing systems that offer similar functionality.
- To identify the requirements of such a system in order to make decent progress in the work and form a basis for future research and implementation.

1.6 Research Questions

The main research question that I will be pondering upon is, "How can the educational system in Nigeria and by extension, the world be improved upon?"

This will involve research into the current, past and proposed educational practices in

Nigeria and other countries and how they can be used for universal academic betterment.

In order to gain the required insights, I will be asking more focused questions in order to find out the cloud and mobile technologies that are already being used in our educational systems and the impact and reach those technologies are experiencing. I will also be looking into how accessible the internet is to students and other stakeholders as well as how they use it.

The research questions are covered in more detail in Chapter Three.

1.7 Significance of the Study

The significance of this study is to improve the lives of people in the country by doing research that will allow me to assuredly propose strategies that when implemented properly will improve the academic condition of the nation. I aim to propose and develop solutions that are user-friendly, accessible, efficient and cost-effective.

1.8 Scope and Limitations of the Study

In this study, I will be focusing on the benefits that could be gleaned in the educational sector from the application of cloud and mobile technologies. More specifically, I will be looking to establish the problems with the current technologies being used and how such issues can be handled. The system that I shall propose will aim to improve on current learning management systems and curtail difficulties students face while in school, learning.

Chapter Two: Literature Review

2.1 Preamble

This chapter of the research includes reviews of research works of different scholars. Moreover, the literature that are reviewed for this section include practical researches in the field of mobile learning and the latest cloud computing technology.

Cloud Computing (CC) refers to the physical structure of a communications network, where data is stored in large data centers and can be accessed anywhere, at any time, and from different devices. This systematic literature review identifies and categorizes the potential and barriers of cloud-based teaching in schools from an international perspective. This study applied the Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA) methodology for determining the articles and reporting analysis. A total of 510 articles were identified from ERIC, IEEE Xplore, Science Direct, and Primo. After screening and eligibility checking, thirteen articles focusing on "cloud computing and school" were included for qualitative and meta-analysis. The papers are coded, devising 31 themes grouped into five categories. The adoption of CC in schools is associated with five factors:

1. Globalization (easier access to education, data access everywhere / accessibility / infrastructure, 21st Century Skills),
2. Educational benefits (motivation, teacher-parent communication, collaboration, flexibility, productivity, creativity, self-organized learning, communication, sharing of knowledge, problem-solving, responsible students/peer review/no bad excuses),
3. Administrative benefits (economy — operations, computer lifetime, licenses, time-saving, BYOD, software diversity),
4. Barriers (technical barriers, cultural barriers, security, privacy, laws and regulations for schools, age restrictions/parent acceptance, and opacity), and
5. Implementation (management support, paradigm shift in education, incongruence between implementation strategies).

Information and Communication Technology (ICT) has led to a paradigm shift in schools in recent years, influencing teachers' roles, learners' roles, content, and practices (Faroese, 2012). ICT is now an integral part of the daily practice of schools around the globe, with teachers being expected to adopt and integrate ICT into their teaching. The Danish Ministry of Education defines the role of ICT as "collecting, processing, storing, and disseminating information" (EMU Denmark's learning portal, 2015, translated). In recent years, the use of ICT in schools has largely focused on the Internet's potential for collaboration, production, and the reproduction of digital products. Specifically, the adoption of Web 2.0 applications and Cloud Computing (CC) has become an essential factor for modern education.

The concept of Web 2.0 first appeared in the article "Fragmented Future" (DiNucci, 1999), but it was made popular in 2004 by Tim O'Reilly and Dale Dougherty at the Web 2.0 Summit in San Francisco. This concept implies a new approach to the Internet,

whereby users work together to create and edit content. Web 2.0 primarily refers to a changed mindset regarding how to use the Internet. On the other hand, CC refers to the physical structure of a communications network, through which data is saved and stored in large data centers and can be accessed anywhere, at any time, and from different devices.

Since 2004, Web 2.0 and CC have made their main entry with services like Wikipedia, Facebook, YouTube, and, later in 2010, with Google Drive, which was a groundbreaking development for its use in education. Jeppe Bundsgaard (2010) divides digital teaching resources into six categories, with CC fitting into the category of shell/utility programs, which are primarily used to produce and store data online. These new Web 2.0 services have revolutionized the way the Internet is used by moving the user from a passive role as the recipient of information to an active role as a player in the content's co-creation. This modified approach to the use of the Internet has spread to schools, where teachers have begun using CC in their teaching. In 2013, 42% of the K-12 schools in the United States were in the process of implementing or already using CC to store data. There was a 15% nationwide increase during the period of 2011 to 2013 (Carahar & Nott, 2013). In Danish schools, many municipalities have chosen to discontinue the use of shared drives in favor of CC. Contrary to this trend, the municipality of Odense, like several municipalities in Sweden, opted not to use Google Drive, as the service did not live up to the municipalities' data processing requirements (Madsen, 2014). It is clear that there are some underlying reasons for the adoption of CC and the discontinuation of alternative technologies, which can be interpreted as the greater potentials of CC and its functions or values that circumvent or even solve some of the existing challenges posed by other forms of educational technologies.

The objective of this systematic literature review is to identify and categorize the potentials of cloud-based teaching in schools from an international perspective. The underlying research question is: from a globalization perspective, what are the reasons behind the integration and adoption of cloud-based applications in teaching activities of schools?

2.2 Literature Review Methodology

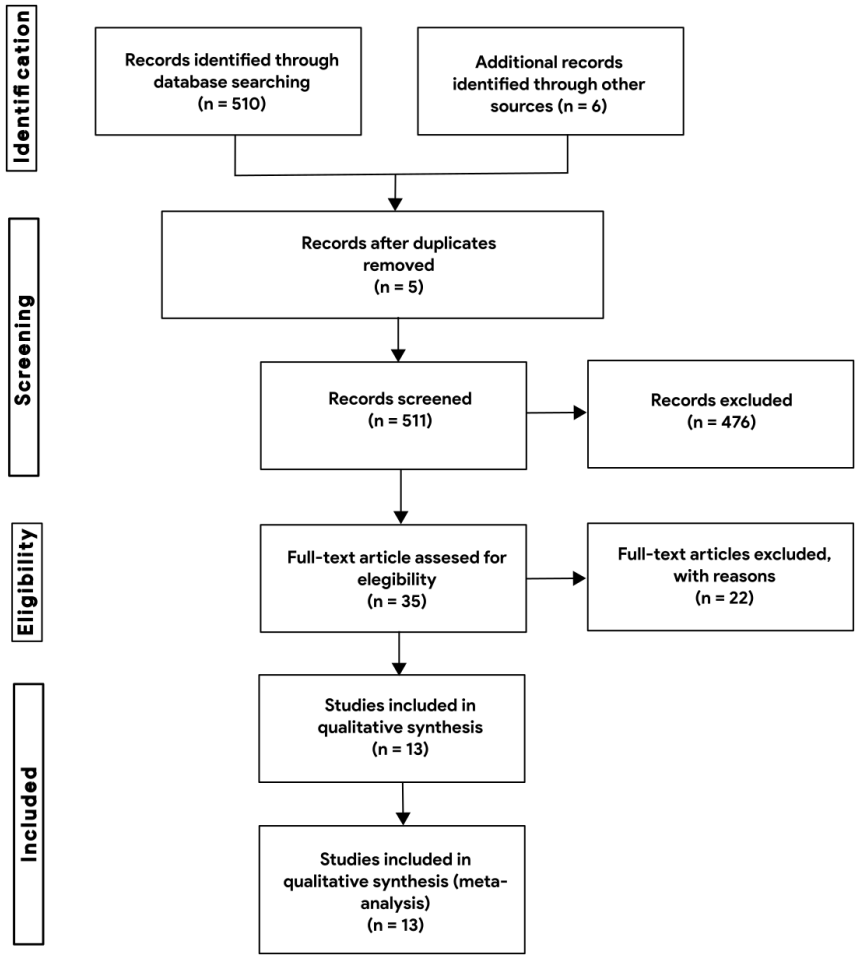


Figure 2.1: PRISMA Flow-Chart

This section provides an overview of the methodological approach used for the literature selection and the method applied for analysis. This literature review is conducted based on the Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA) methodology, which is used to create a systematic, precise, and reliable overview of the literature (Moher, Liberati, Tetzlaff, & Altma, 2009). The process of collecting articles for analysis is guided by a flow-chart that includes four phases, and the process for reporting a literature review includes a 27-step checklist. PRISMA defines a systematic review as "a review of a clearly formulated question that uses systematic and explicit methods to identify, select, and critically appraise relevant research, and to collect and analyze data from the studies that are included in the review" (Moher, Liberati, Tetzlaff, & Altma, 2009).

This paper reviews the potential of cloud-based applications in schools. Therefore, various combinations of the following keywords were used: cloud, K-12, classroom, benefits, pros, potentials. The search was restricted to peer-reviewed texts and was conducted in English to obtain a global perspective on the topic and avoid too narrow searches. The search was

limited to the period between 2010 and the search date (September 24, 2018). This restriction was implemented because Google Docs, a CC front-runner in education, went from the beta version to the official in 2010. Furthermore, in 2010, the arrival of iPad revolutionized the development and adoption of could-based educational applications through different platforms. In consultation with a librarian and after testing different combinations of keywords, four databases were selected, and different combinations of the keywords were used. ERIC returned 23 results, IEEE Xplore returned 266, Science Direct returned 171, and Primo returned 50.

In Figure 2.1, the PRISMA flow diagram, shows four phases: "identification," "screening," "eligibility," and "included." The "identification" phase returned 510 results via the four databases. In addition, six texts were included as the result of chain searches, which is where known literature is used as a starting point for finding additional literature related to the topic. In the first part of the "screening" phase, five duplicate articles were removed. Furthermore, 476 texts were excluded because the title, keywords, or abstract did not contain the desired themes. The criteria used to reduce the number of texts were as follows:

1 The text has no connection to schools and
2 The text does not refer to any kind of cloud service.

In the third phase, "eligibility," 35 texts were left, and of these, an additional 22 texts were excluded because they did not contain a sufficient connection between schools and cloud services. In the final phase, "included," there were thirteen texts remaining to be included in the quantitative synthesis and analysis.
This study applied a coding technique presented by Tayler-Hewitt (2001). The articles were encoded according to themes and then divided into categories. During this process, the coded sections were regularly compared to similar parts of texts containing the same codes. The intention was to create a connection between the texts and ensure the continuity of the codes' definitions (Strauss & Corbin, 1990; Hewitt-Taylor, 2001).

2.2.1 Globalization in the Context of Education

Globalization in the education context includes a number of intertwined discourses and processes worldwide. Spring (2008) refers to globalization and education as involving discourses concerning knowledge, economy, technology, and lifelong learning. Such factors in a global context influence a variety of practices in a local context. Under the influence of major intergovernmental institutions, such as the OECD, the global education policy forms a so-called "super structure" through various international networks and processes that influence the school system at a national level. The authors also talk about a global super structure, which refers to when multiple nations adopt a political agenda on the basis of a global policy in order to comply with the global economy. For example, Denmark's National Learning Festival, held in September 2015, put a special focus on 21st century skills. These skills were originally formulated by global public and private organizations in 2002 in order to create a learning model that would prepare students and citizens for the new millennium, where the development of these skills would support international competitiveness (P21, 2015). Spring (2008) expands on this by examining the extent to which the world's governments discuss how to invest in training for human capital development for the sake of promoting economic growth, saying that, "As a consequence, educational discourses around the world often refers to human capital, lifelong learning for improving job skills, and economic development" (Spring, 2008,

p. 132). He also accounts for technology's role in this context, explaining that, "Information and Communication Technology is speeding the global flow of information and creating a library of world knowledge" (Spring, 2008, p. 132). This paper thereby considers the term globalization as a product of the discourses and trends of international and intergovernmental institutions that influence the national school practice for the sake of developing training and human capital that can promote global competitiveness. At the same time, ICT acts as a generator of this development.

2.2.2 Competence

The term competence can be defined as a complex fusion of knowledge, skills, and abilities, and the use of these in a given context. The European Network KeyCoNet (2012) selected eight key competences that were considered necessary for achieving personal development, active citizenship, and social inclusion in the 21st century. These included digital, social, and civic competences and the skill to communicate in the mother tongue (Looney & Michel, 2014). The educational focus has shifted from teachers' teaching approaches to students' learning approaches. Additionally, competence is described as the context-oriented skills of a student—namely, the skills that make him or her able to cope in specific situations in real life and capable of working in certain professional fields. Competence is even formulated as concrete learning. For example, the students' ability to solve problems would be the goal (Egelund, 2014). Therefore, one of the objectives of this paper is to explore what the existing literature says about the use of cloud services in this characterization of 21st century competency.

2.2.3 Lifelong Learning and Educational Development

Spring (2008) presents lifelong learning as a part of the discourse on the knowledge economy. Since technology changes rapidly and continuously, there is a continuous change in the knowledge economy. Therefore, it is important to prepare students for lifelong learning. This focus has a substantial impact on curricula (Spring, 2008). It is not just the technologies that are constantly changing, but also the global job market, and that is why Spring (2008) discusses the essential skills that students have to learn, namely: communication, interpersonal, and collaboration skills; the capacity to assume personal responsibility; and the ability to learn new subjects. At the same time, Castell presents the concept of "self-programmable labor" (Levinsen, 2011), which refers to the labor in the network society that allows self-initiated lifelong learning. This type of labor, unlike "generic labor," succeeds in the context of global competition. Castell suggests that the education system must accept the task of fostering self-programming students (Levinsen, 2011). Thus, in this study, one of the viewpoints is to inquire as to how CC contributes to lifelong learning and educational development.

2.3 Qualitative Synthesis

Figure 2.2: Geographical distribution of the articles

This section presents the qualitative synthesis of the thirteen articles on using cloud computing in schools. Figure 2.2 above shows the countries where the literature originated from.

Table 2.1 below presents an overview of the articles and a matrix for the identified themes in each of the papers.

Table 2.1: Schematic overview of the articles

Year of Publication	**2010**	**2011**		**2012**				**2013**					**2014**
Author	Thomas	Weil	Carroll, Merwe & Kotze	Chandra and Borah	Stein, Ware, Laboy & Schaffer	Bedell, C.	Masud, Xiaodi & Yong	Alixadeh & Hassan	Hung & Lin	Mahalingam & Rajan	Konvot, Norton & Cloutier	Ullman	Bathon

Title	A potential paradigm for practicing the scholarship of teaching and	On Cloud Nine	Secure cloud computing: Benefits, risks and controls	Cost benefit analysis of cloud computing in education	Improving K-12 pedagogy via a cloud designed for education	The state of computing in K-12	Cloud Computing for Higher Education: A road-map	Challenges and opportunities of mobile cloud computing	An integrated cloud application in school knowledge management implementation	Cloud and mobile computing: Affordances of the 21st century teaching and learning	Privacy and cloud computing in public schools.	The case for cloud computing in k12	The fine print on cloud computing
Background													
Geographical distribution	Botswana	USA	South	India	USA	USA	Australia	Malaysia	Malaysia	Dubai	USA	USA	USA
Type of text: Conference article (C) or journal article (J)	J	J	J	C	J	J	C	C	C	C	J	J	J
Themes													
21st century skills	x			x					x	x			
Age Restrictions/parents' acceptance											x		
Responsible students/Peer review/No bad excuses	x	x											
Bring Your Own Device (BYOD)		x		x		x			x	x		x	
Cloud implementation		x		x			x		x		x	x	
Data security	x		x		x	x		x			x		x
Data access everywhere/accessibility/infrastructure	x	x			x	x	x	x	x	x	x	x	
Software diversity		x			x				x				
Documentation	x	x							x				
Reliability											x		

Flexibility	x	x		x		x	x	x	x	x		x	
Globalization	x	x		x		x			x	x		x	
Incongruence between implementation strategies							x		x				
Collaboration	x	x		x		x			x	x		x	
Communication (students/teachers/parents)		x									x	x	
Creativity				x						x			
Cultural barrier		x			x	x	x		x	x			
Management support									x	x			
Laws and regulations for schools	x										x		x
Motivation	x	x											
Easier access to education	x					x							
Paradigm shift in education	x	x				x						x	
Problem-solving										x			
Productivity	x	x		x		x				x		x	
Self-organized learning		x				x							
Teacher-parent communication													
Technical barrier					x			x			x		
Time saving	x											x	
Opacity											x		
Sharing of knowledge	x	x				x	x			x		x	
Economy (operation, computer lifetime, licenses)	x	x			x	x					x	x	

To begin with, the selected literature was read and the relevant concepts were coded. The codes were generated from the text material. Accordingly, the analysis has an inductive starting point (Hewitt-Taylor, 2001). These codes represent different themes associated with the overall subject. A total of 31 themes were generated by the codes to reflect the context of the 13 texts. These 31 themes were then synthesized under five broad categories, which cover all the themes. Subsequently, all texts were read again, and the themes, in the form of selected quotes, were distributed across the five categories. Finally, the citations were processed according to whether they were central points, convergent statements, or divergent statements. The final qualitative synthesis was performed on these five categories. Hewitt-Taylor's (2001) approach was regularly inspected to obtain the criteria: reliability, credibility, and transparency (Lincoln & Guba, 1985; Hewitt-Taylor, 2001). The themes were distributed across the following categories:

- **Category 1: Globalization**
  Easier access to education, data access everywhere / accessibility / infrastructure, 2-century skills

- **Category 2: Educational benefits**
  Motivation, teacher-parent communication, collaboration, flexibility, productivity, creativity, self- organized learning, communication, sharing of knowledge, problem-solving, responsible students/peer review/no bad excuses

- **Category 3: Administrative benefits**
  Economy (operations, computer lifetime, licenses), time saved, BYOD, software diversity, documentation

- **Category 4: Barriers**
  Technical barriers, cultural barriers, data security, privacy, school laws and regulations, age restrictions/parent acceptance, opacity, reliability

- **Category 5: Implementation**
  Management support, paradigm shift in education, incongruence between implementation strategies, cloud implementation

2.3.1 Globalization

Some of the most striking examples of CC's influence in the global arena is described in the articles at two levels. One example is the use of CC in an educational context to increase living standards in the world's most vulnerable regions. The other is the inherent potential of CC to support the skills needed in the global community.

Mahalingam and Rajan (2013, p. 125) give insight into how CC has found its way into education, explaining that, "One such development which is gaining acceptance and widespread adaptation is 'cloud computing,' supported by the notion of 'bring your own device' (BYOD)". The prevalence of CC in areas of great economic diversity is due to their perspective, which is partially the result of BYOD, as this enables the areas to be included in the global education community. Chandra and Borah (2011) adds that there is less of a need to upgrade the software when using CC.

India is described as an example of how education is becoming an important factor for development and the rise in general living standards:

- The development of the education sector is [a] solution for economic growth and improvement in the standard of living. The challenges posed by the growing demand for education requirements are gigantic. India will have about 45 million people in the age group of 18 years to 20 years by 2020 (Chandra & Borah, 2011, p. 4).

Therefore, India wagered on the National Knowledge Network (NKN), whereby the country's knowledge-related institutions were given high-speed networks to make scientists and researchers capable of offering students instructions via CC in areas of growth or deprivation (Chandra & Borah, 2011). Bedell (2014) estimates that CC helps ensure that people in all regions can have access to quality education, something that cannot be done without the cloud. Therefore, in the coming years, CC is predicted to grow globally in its use for educational purposes.

Duffy and Jonassen (in Mahalingam & Rajan, 2013) look at the demands of globalization for the students' skills in the 21st century and how teachers, by utilizing CC, can support the development of these skills among students: Teachers in the role of facilitators can leverage the cloud infrastructure built by Google, Box, Podio, Microsoft and several others to practice and prepare the students for the 21st century skills. Students can now create a project report in Google Drive or Microsoft SkyDrive and start collaborating by editing, commenting and engagement in knowledge construction any time anywhere working simultaneously in the same document, which changes the way people look at collaboration in the 21st century (Mahalingam & Rajan, 2013, p. 70).

The inherent affordances of CC that enable a ubiquitous global learning environment can facilitate an educational paradigm shift from teacher-centered to student-centered learning. These changing circumstances require teachers to be better prepared for new technology. CC's potential in education is further elaborated in the following section.

2.3.2 Educational Benefits

The literature points to a number of advantages for CC use in education, some of the most prominent being collaboration, efficiency, motivation, universal data access, and unlimited space for data storage.

The widespread use of BYOD in schools, as well as the increase in the use of tablets in the classroom, supports the need to consider CC (Mahalingam & Rajan, 2013). One of the major benefits of CC in education is the potential for collaboration. As Chandra and Borah put it, "In Cloud Computing [,] teachers and students [as well as] students and students can implement collaborative learning such as the online exchange, online document editing, … [and online use of the] concept map tool[s] like [the] Google Collaboration Platform" (2011, p. 2). The intent of collaboration is for all students to contribute, which helps to create a cooperative culture (Hung & Lin, 2013; Thomas, 2011).

Collaboration creates a community in which it is possible for students to receive emotional support while building trustful ties within the student group. This is not restricted to inter-student relations, however, as teacher interactions can also be improved through CC. As (2011) described, "If Mitchell happens to be online while students are working on their assignments, he can simply ‘pop in.' Students can communicate with him or with other students via instant message. Mitchell can also give instant demonstration, and the students can follow up immediately with questions" (p. 3).

This kind of "real time" feedback is crucial to students' success. The fact that teachers can collaborate on the knowledge construction with their students enables them to give timely support and comment on their work (Mahalingam & Rajan, 2013). The community this creates will be capable of overcoming the indifference of self-studying and improving interactions between the users (Chandra & Borah, 2011; Hung & Lin, 2013).

Furthermore, cloud computing also creates more efficiency and motivation among students. "Using the application also holds students more accountable; excuses like ‘the dog ate my homework' don't fly in the virtual world [...] Students come to class prepared […] That alone creates a more enriching and efficient classroom environment" (Weil, 2011, p. 3). With this technology, teachers can teach more convincingly, and thus create greater enthusiasm and motivation among students (Weil, 2011).

Another major advantage of CC is that it provides access to information anywhere. As Thomas puts it, "Users can reach into the cloud for resources as they need from anywhere at any time. For this reason, cloud computing has also been described as 'on-demand' computing" (2011, p. 218). This creates more flexibility in the school day and allows learning to continue outside of school (Stein, Ware, Laboy, & Schaffer, 2012). The primary advantage of this is that teachers and students have easy access to information and can share documents immediately. This means that we are moving more and more into a future of self-organized learning, according to Bedell (2014). Furthermore, the availability of information outside of school improves communication between school and home:

- Through access to certain information stored in the cloud, teachers can have better communication with parents and students when it comes to projects, tests and assignments. Teachers can post messages. Parents can log in to secure servers to check their child's progress and online cloud forums allow for two-way school-to-home dialogues (Ullman, 2015, p. 3).

By extension, the literature highlights unlimited storage in cloud services as an advantage. "Both Microsoft and Google recently announced unlimited storage, relieving districts of the massive storage requirements associated with digital student portfolios" (Bedell, 2014, p. 3). The advantage of this is that no one needs to make backups or transfer data from one device to another (Thomas, 2011). Groups can also quickly gather a variety of information in one place and easily share it with another group for viewing or editing, the whole school, or the entire world (Hung & Lin, 2013). The same applies to the teachers:

- Teachers can review what they have accomplished in the past. At the beginning of the year, when teachers make their plans for the upcoming school year, it can serve as a good repository of their experiences and ideas, especially, when different personnel take over new posts. Moreover, they can quickly review past experiences and work tasks (Hung & Lin, 2013, p. 7).

2.3.3 Administrative Benefits

At the administrative level, CC distinguishes itself by offering a number of benefits to the individual school, the whole school district, and collaboration across schools. The administrative benefits can be divided into:

- The economic benefits of the use of CC;
- Division of labor, which can be improved by using CC; and finally,
- The streamlining of already established school procedures.

2.3.3.1 Economy

The literature paints a clear picture of a wide range of economic benefits for using CC in schools. For one thing, CC can serve as a more cost effective way to increase productivity and collaborative learning activities in schools (Bedell, 2014, p. 4).

By using CC, there is no longer a need to install software locally on the computer, which is preferable economically since software maintenance is outsourced to the cloud service. As Thomas (2011) describes, "There is no need to buy hardware and software

licenses and pay for maintenance. Thus, the cloud model offers a much cheaper way to acquire and use IT services; this is quite beneficial especially for educational institution in these days of appalling economic crunch" (p. 218). In addition to the maintenance savings, CC allows document exchange between teachers and students to be done digitally, which eliminates the need for expensive printing and copying solutions. Ullman (2015) estimates that switching to a cloud service saves up to five times more than the current analog solutions.

The use of CC is also highly scalable, as it is both quick and easy to purchase additional licenses for a cloud service, as opposed to investing in new software installation. It is easy to start small and then buy more licenses when the need arises (Ullman, 2015, p. 6). Furthermore, "By its design, cloud computing is scalable, flexible and elastic—offering IT department a way to easily increase capacity or add additional capabilities when necessary, without investing in new and expensive infrastructure, training new personnel, or licensing more software" (Thomas, 2011, p. 219; Stein, Ware, Laboy, & Schaffer, 2012). In many cases, different users can use the software licenses at different times. For example, a school of 500 students could buy just 100 software licenses to cloud services and then take turns using them. As Ullman (2015) explained, "With most cloud services, the district pays for only what it uses (in terms of storage and services)" (p. 5). In many cases, a higher price is negotiated for shared licenses to compensate for the increased use, but it is still a significant saving for financial pressed schools (Stein, Ware, Laboy, & Schaffer, 2012, p. 238).

Cloud services are used online, so there is no software installed locally on the school's computers. This releases storage space and allows older computers an increased their life span, as the cloud services are run remotely from a computer with better processing power. As one piece of literature describes, "The Cloud opens the door to low-cost computing power. Because the computational power needed to run applications is provided remotely, schools can save costs through the use of older and less powerful computers. Lengthening the life of hardware substantially lowers the cost of ownership" (Stein, Ware, Laboy, & Schaffer, 2012, p. 238). Chandra and Borah (2011) conclude that the cost of switching to a cloud service is cheaper than replacing older computers.

2.3.3.2 Division of Labor

By switching to CC, technical barriers such as software updates and system crashes are no longer an issue for the school or the teacher. The cloud service provider ensures that the software is updated and working properly and allows further access to the material anywhere and anytime (Weil, 2011, p. 4). This frees up the teachers' time as they no longer need to take care of technical challenges, but can instead focus on their core mission—teaching (Hung & Lin, 2013; Ullman, 2015).

2.3.3.3 Streamlining

Teachers can be more efficient by using CC, since "by storing content online, teachers no longer need to spend time and resources printing or copying long documents or lesson plans" (Ullman, 2015, p. 3). In addition, the barriers that some teachers face by working with technology disappear, as working with CC does not require special IT knowledge from the teacher, as "there is no requirement for professional programming abilities. We do not need to worry if the School does not have a professional programmer. We can

concentrate on our work and responsibilities as long as we can do basic data entry" (Hung & Lin, 2013, p. 7).

Teachers' and students'materials are stored online, which eliminates both the need for USB drives to store data and the need for archives to hold large folders with printed assignments and student information. Thomas (2011, p. 219) describes efficiency as "no need to copy all stuff from one PC to another when buying a new one. It also means you can create a repository of information that stays with you and keeps growing as long as you want them."

2.3.4 Barriers

After a review of the educational and administrative benefits of CC, it is important to clarify what barriers the literature highlights. The cultural barriers are most important, while the technical barriers have a less prominent role.

Besides these two barriers, there are also a number of concerns regarding the use of CC in schools. These uncertainties mainly regard confidential agreements, which, among other things, involve privacy, security, anonymity, monitoring, reliability, and responsibility (Thomas, 2011, p. 220).

2.3.4.1 Cultural Barriers

As the literature describes, "One of the central insights we have gained is that the barriers to adoption are as much, if not more, cultural than technical" (Stein, Ware, Laboy, & Schaffer, 2012, p. 239). Cultural barriers are one of the biggest obstacles to the integration of CC. One of the critical challenges is the mismatch between digital immigrants, who constitute the majority of teachers, and the digital natives, or the students. The challenge for teachers is to manage and follow technological development (Mahalingam & Rajan, 2013).

Much of the resistance to change to adopt these technologies is intrinsic rather extrinsic. Teachers are often comfortable to main the status quo and resist changing due to the fact that these technological affordances are still considered as optional and often viewed as tech savvy teacher's prerogative (Mahalingam & Rajan, 2013, p. 127).

Teachers must see the benefits of the technology and discover that they can meet their needs more efficiently by using it. In addition, there needs to be training and support in the use of CC from a colleague or an IT professional who can see the potential and is willing to carry the initiative forward (Stein, Ware, Laboy, & Schaffer, 2012). It requires planning and leadership to train these teachers so they can see the value in CC, understand how to use it, and integrate it into their current pedagogy (Bedell, 2014). The potential for sharing knowledge effectively among teachers via CC is obvious, but an organizational culture that supports this method of working must first be established, as this will create a change that supports and encourages teachers to use the technology so empty virtual archives can be avoided (Hung & Lin, 2013, p. 8).

Another barrier is the fact that technology changes rapidly, meaning that once the teachers have learned to use one function, it either evolves or a better competitor enters the market. According to Mahalingam and Rajan (2013), this creates confusion and frustration among teachers, which means that they end up waiting to see what the next change will be and the

integration of technological development fails.

2.3.4.2 Technical Barriers

Previously described under administrative benefits, the technical barriers are no longer a challenge for schools, as the providers of cloud services are responsible for updating and maintaining the software. Additionally, it is required that the Internet functions, including its speed, so the practitioner can use CC, and each device must allow this use of it (Alizadeh & Hassan, 2013; Mahalingam & Rajan, 2013, p. 12).

2.3.4.3 Security

The literature highlights data security as a major concern, since schools have an obligation to treat sensitive student data confidentially (Stein, Ware, Laboy, & Schaffer, 2012, p. 238; Bedell, 2014, p. 4). Barthon (2013) points to this particular issue as a major limiting factor for schools to fully start using cloud services.
In a digital world, data is constantly created, archived, and deleted, and the default setting for the Internet is that it is open, meaning that all data that interacts with the Internet can be shared. This creates a problem for schools, which are under an obligation to keep student information secure (Bathon, 2013, p. 23). For this reason, confidentiality agreements between these cloud services and schools or school districts are crucial (Bathon, 2013, p. 24). As Bathon (2013) phrased it, "It is important that these documents specify that the private cloud company has no ownership interest in the intellectual property contained in the uploaded data" (p. 24).

However, cloud services can be difficult to fully understand, and the confidentiality agreements are no exception. One US study showed that only 25% of school districts inform parents that they use CC, and 20% of districts have no policies regarding online services. In addition, a large proportion of confidentiality contracts contain large gaps or are completely lacking in privacy policies (Reid Berg et al., 2013, p. 5).

This survey also indicated that school districts often surrender control of student information when using cloud services. Less than 25% of the agreements between school districts and cloud services specify the purpose of the publication of student information, and fewer than 7% of the districts prohibit the sale or use of student information for commercial purposes. Despite this, the Family Educational Rights and Privacy Act (FERPA) requires that the districts have direct control over the students' information when they are published to a third party (Reidenberg et al., 2013, p. 5).

2.3.5 Implementation

Regarding the implementation of cloud services in educational institutions, the literature presents a number of recommendations, primarily aimed towards considerations related to the chosen implementation strategy. It also gives recommendations in order to clarify and anticipate the various barriers that can occur when CC is used in education.

In preparation for a strategic plan for CC implementation, it is recommended to identify the motives for the change: "Drivers for considering cloud computing solutions for educational institutes should be identified in the context of the institutional strategy and how well they align" (Masud, Yong, & Huang, 2012, p. 555). In order to succeed, it is important

that the institutional strategy adapts to the CC needs:

- The success of the strategy implementation depends on the existence of a service-oriented architecture at the level of the institution that offers the necessary infrastructure for cloud implementation. Also, in order to have success, the cloud strategy must be aligned with the institutional strategy (Masud, Yong, & Huang, 2012, p. 553).

Furthermore, it is recommended that this be done early in the process in order to fully benefit from the experiences of pilot projects and to identify solutions to challenges before the full rollout is implemented. Hung and Lin (2013, p. 7) added another relevant consideration: "The most important consideration is the needs of its users." In this perspective, the user requirements should be the main motive in the implementation strategy. Once a school district has decided to implement CC in education, it is important that it establishes policies and implementation plans for the use of the cloud service, according to Reidenberg et al. (2013). Teachers need training courses, and there must be readily available support to provide advice and assistance to teachers in the handling of data and security. Thomas (2011) points out that most teachers who make use of educational technology simply convert educational content into an electronic format and preserve the traditional knowledge centered teaching didactics. This may delay the implementation process considerably, as: "the change is very slow or not forthcoming at all for various reasons" (Thomas, 2011, p. 216).

Moreover, the support of school leaders determines whether a new technology such as CC is integrated in the classrooms and used by the teachers (Hung & Lin, 2013). It is important that these leaders promote the importance of knowledge sharing and the creation of a genuine knowledge-sharing culture among teachers.

2.3 Quantitative Synthesis

Figure 2.3: Word cloud of the keywords (generated using www.wordle.net)

As a basis for the qualitative analysis, the remaining thirteen items were used and described

in the chart below (Figure 2.3). Five of these articles were presented at conferences between 2012 and 2013, and the remainder were published in journals between 2010 and 2015. Six of the thirteen articles describe the conditions of the use of CC in the American school system. Furthermore, articles are included from Botswana, Malaysia, India, Dubai, South Africa, and Australia.

The authors of these articles represent great geographical diversity. The majority of the researchers are based in the universities of the United States, including the Massachusetts Institute of Technology, Fordham University, and the University of Kentucky. Beyond that, researchers are affiliated with universities in Australia, China, Japan, Malaysia, Botswana, South Africa, and the United Arab Emirates. Therefore, the articles present a wide geographical distribution of contexts for CC use, which enables a globalized look at the use of CC in schools.

In Figure 2.3 is a representation of the keywords that are used most frequently in the 13 articles. This word cloud creates an immediate overview of the essential themes from the articles.

2.4 How Education Has Evolved

The field of education has evolved rapidly over the years. The advent of various Information and Communication Technologies have brought many changes to the field of Education.

At the beginning of the century computers were not even used in schools. They did not even exist in their current form. Since the 1980s when computers began to be seriously used to aid learning the benefits have been massive. Networking technologies have brought schools online, allowing research and learning material to be accessed via the internet. Interactive learning tools has also increased the level of automated learning in the educational system.

More recent developments such as cloud technologies have made it possible for educational material to be interacted with from anywhere in the world from any device that has a browser.

Ubiquitous mobile technologies highlighted by the smartphone allows people to access educational content whenever they are free and wherever they are. Progressive mobile technologies allow educational material to be smartly cached. This means that when users are connected to the Internet the learning material can be downloaded and saved to storage, and if they go offline, they still have access to the content that has already been downloaded.

2.5 Benefits of the Proposed System

The system I propose offers the following benefits;

- **Cloud Based** – The fact that the system is cloud based allows learning materials to be stored in the cloud and accessed from any device that can run a browser.
- **Mobile Based** – The front-end of system shall be tailored to mobile. The benefit that this brings over traditional web based solutions is that people carry their phones everywhere they go. This means that they will be able to access their learning

material everywhere they go as well.

- **Progressive** – The concept of progressive technologies is that application is not mortally hamstrung when there is no Internet connection. Applications employing this progressive methodology, download all the data and files needed to run the application suitably to local storage (in this case the learning content). After these files have been downloaded the application is able to work appropriately whether there is an internet connection or not.
- **Data Cost Friendly** – The high cost of Internet access in developing countries prevents many people from using Internet enabled technologies to learn. The proposed system shall minimize data usage by compressing files before transferring them over a network. The front-end of the system shall also use P2P technology to allow users share learning material with one another without using any data.

2.6 Conclusion

The purpose of this paper was to investigate and provide an overview of the current cloud technologies in use in educational systems around the world. Using the systematic review methodology, thirteen articles dealing with the integration and adoption of cloud computing in schools along with the benefits and shortfalls were selected and analyzed. These articles cover experiences across seven different countries and four different continents. After a thorough reading of the texts and the use of coding techniques, 31 themes and 5 categories were created as the basis for a qualitative analysis.

This paper has uncovered two distinct potentials regarding the use of CC, from a globalization perspective. First, CC allows teaching and learning to be accessed by a larger part of the population, including in countries and regions where distance learning has thus far been problematic. This increased level of education creates economic growth and can thereby increase the living standard in the world's most vulnerable areas. Second, the use of CC supports and scaffolds the current global discourse on educating future citizens with 21st century skills. However, there is still a gap in the literature showing the geographical distribution of CC use worldwide.

On a local level, there are a number of advantages in education related to the use of CC in schools. The prominent advantage is that CC enhances student collaboration, which promotes motivation and helps keep the students responsible for their own work within the community. The universal access to data across time, space, and devices allows the teaching, learning, and collaboration to take place anywhere and at any time. Finally, the review points to the advantage of unlimited data storage, which removes the problems of data loss or inaccessibility; these functions enable students and teachers to develop an ongoing portfolio of the students' work and progress in school.

A number of administrative advantages are also identified, especially regarding the use of CC in schools and across entire school districts. Among these are a number of economic benefits that the educational institution or the district can harvest by discontinuing local storage and authority and by adopting cloud service. These include maintenance outsourcing, extended hardware life, reduced licensing costs, and savings on expensive printing and copying solutions. In addition to the economic aspect, a number of efficiency improvements exist in the organization of work, which can reduce local resource requirements.

Despite the clear potential, the literature has highlighted a number of barriers in

implementing CC in schools. Local cultural norms can inhibit to change, and the disparity between the teachers as digital immigrants and the students as digital natives presents an issue. Safety is also a challenge that cannot be neglected when using CC in schools. Schools are subject to national guidelines for data security, and students have the right to the privacy and the security of their data. There is a sharp contrast between the characteristic openness of the Internet and the impenetrable confidentiality agreements that the users and providers of the cloud services require. The literature is still incomplete regarding the security issues, and the results presented in this paper are exclusively based on experiences from the US.

Finally, some recommendations are presented for the future implementation of CC in schools. This includes a clear implementation strategy and consistent management support. Moreover, the review presents a great deal of advice for clarifying and anticipating the barriers that can occur when a CC system is implemented in a school.

It is clear that CC should be implemented with careful consideration of the factors associated with each school's context.

Chapter Three: Data Collection, Analysis and Findings

3.1 Preamble

In the previous chapter, a review of the academic literature associated with cloud computing was carried out. This formed the foundational knowledge needed to understand this research. This chapter focuses on addressing the second subjective which is gathering all necessary requirements for the learning material management system.

In order to get the current way of managing course material, a detailed study of the current process is carried out. The key observations about this current process will provide us specifications of the new system.

This chapter begins with a brief description of the case study and current process employed in managing learning materials and other education applications. Next, method of data collection will be addressed and data collected will be analysed and results will be generated. The result will lead to several key findings. These will form the basis for establishment of the user requirements for the new system.

The purpose of this research project is to develop a learning material management system that can be effectively used in the learning process. The aim is to synthesize solutions to help to educational institutions to make better use of mobile and cloud technologies whether it is by proposing ways in which the learning and teaching processes can be streamlined or by helping students to study better when out of the classroom. This chapter covers the data collection, research design and methodology, including sampling, population, establishing rigor during and after data collection, ethical considerations and data analysis.

3.2 Research Problem

The central research problem revolves around the issue of how schools in Nigeria can make better use of available cloud and mobile technologies in a way that is helpful to all stakeholders of the academic value chain. It is common knowledge how the aforementioned technologies have helped boost productivity in other sectors of the economy both nationally and internationally, but it largely remains to be seen how they can help schools across the country in a manner that is both effective and affordable for the institutions in question. We will also be looking at the current technologies already being used by students. These are the insights which this research aims to unearth.

3.3 Research Questions/Hypotheses

The questions that I will be seeking to find answers to are as follows:

1. Basic demographic information on the respondents including their gender, age groups, study level (undergraduate, postgraduate, or research), faculty or discipline, educational institution type.

2. What kind of devices do the respondents have and make use of?
3. What operating systems run on these devices?
4. Do they have reliable access to internet on these devices?
5. What tasks do they use the internet to perform?
6. How aware are they of cloud or mobile learning applications?
7. How much time do they spend on these cloud or mobile learning applications?
8. Which type of cloud or mobile learning apps do the respondents use?
9. How effective do they find these applications in helping them learn?
10. How impactful are these mobile applications to their education?

The questions above range from open-ended, to yes or no, to scale of preference (Likert scale).

Please note that a some of the applications mentioned here were not specifically developed to be used as educational applications but their overall utility functions allow them to be used for such.

3.4 Methodological Approach

When faced with the choice of picking between a qualitative or quantitative approach, I opted to go the route of a quantitative one. The reasons I went with the quantitative approach were as follows:

1. The need to get responses from as diverse a group as possible.
2. The need to get responses from a large enough sample size since my solution will aim to help as relatively large audience.
3. I was looking for a very cost-effective and time-efficient approach.

3.5 Research Instruments

For this research, the instruments I used for the survey were, questionnaires that I distributed online. These questionnaires were used to obtain information about the current state of mobile and cloud learning technologies.

The questionnaire comprised three sections and was distributed to people online (via Google Forms). The first section (Section A) focuses on the user's demographic profile, considering details such as gender, age, and educational background, affiliation and discipline. The second section (Section B) covers students' awareness and usage of mobile and cloud technologies via the Internet. The final section (Section C) investigates the awareness and usability of mobile learning in higher education.

For some questions a 5-point Likert scale anchored by "Strongly Disagree" (1) and "Strongly Agree" (5) was used. The ranges of five point Likert scales were categorized into equal sized categories of low, moderate, and high.

3.6 Description of the Existing System

The existing system is Moodle, an open-source learning management system being used by the University of Jos and many other universities all over the world. Moodle is an on-premise web based application that is deployed on the institutions server and accessed on a website.

On-Premise (also known as on-prem) software is installed and runs on computers of the person or organization that is using the software, rather than a remote facility such as a server farm or the cloud.

3.7 Analysis of the Existing System

Often, the purpose of developing a new system is to replace an existing one, as is the case here. Hence, there is much to gain from analyzing the current system in order to gain insights into the problems being faced. A proper analysis will ensure that the right parts or characteristics of the system are upgraded or modified.

The main characteristics of Moodle are as follows;

- **Web Focused** – Users interact with Moodle via the browser.
- **Online Oriented** – Moodle works only when the user is connected to the internet. This means that when users are not connected to the internet and have no access to the University's local network, they are unable to access their course material if they had not downloaded them previously.
- **Desktop Focused –** The Moodle UI was designed almost solely with Desktops and big (13+ inch) screens in mind. This means that when the web application is visited on a mobile device, the user interface is found wanting due to its lack of responsiveness. This poses a problem as a large number of users access the web from their mobile devices and shall receive a user experience that is below par.
- **On-Premise** – Moodle runs on the servers of the University. This means that if there is a power outage or unfortunate event at the server location the whole system will go down. No matter how good the University's Infrastructure or Dev Ops processes are, a reputable cloud provider will almost always be a better bet.

3.7.1 Poor Uptake by University of Jos Students

As I previously alluded to, Moodle is being used by the University of Jos. The fact of the matter is that the only department that makes nearly adequate use of Moodle is the Department of Computer Science. This is simply not good enough.

With my system, I aim to remedy this issue and boost usage of learning platforms in the school.

3.8 Methods of Data Collection

A survey was used to obtain information about the current states of mobile based cloud learning technologies. The questionnaire comprised three sections and was distributed to people online. The first section (Section A) focuses on the user's profile such as gender, age, and educational background, affiliation and discipline. The second section (Section B) covers students' awareness and usage of mobile and cloud technologies via the Internet. The final section (Section C) investigates the awareness and usability of mobile learning in higher education. For some questions a 5-point Likert scale anchored by "Strongly Disagree" (1) and "Strongly Agree" (5) was used. The ranges of five point Likert scales were categorized into equal sized categories of low, moderate, and high.

3.9 Data Analysis and Results

3.9.1 Respondents' Profile

As shown in Table 3.1, 57.86% of the respondents were male and 42.14% female. The most (45.36%) of respondents were undergraduate students, 37.86 were postgraduate students, and 16.78% were enrolled as PhD. students. The majority (59.64%) of the respondents were aged between 21-30 years. Medical Science students made up the smallest group of respondents with 3.93%, while respondents studying Natural Science courses made up a plurality of 32.14%. In terms of affiliation, most of the respondents (64.58%) were from the University of Jos and the remainder (35.42%) were from other universities. This data indicates that the findings represent opinions of different levels, age and disciplines of students from different universities.

Table 3.1: Demographic Profile of the Respondents

Profile	Classification	Total (N = 280)			Percentage (100%)
		Male - 162 (57.86%)	Female - 118 (42.12%)	Cumulative	
Study Level	Undergraduates	72	55	127	45.36
	Postgraduates	58	48	106	37.86
	PhD students	32	15	47	16.78
Age Group	Below 21 Years	45	38	83	29.64
	21-30 Years	96	71	167	59.64
	Above 30 Years	21	9	30	10.72
Faculty or Discipline	Natural Sciences	42	38	90	32.14
	Social Sciences	31	32	63	22.50
	Medical Sciences	9	2	11	3.93
	Law	12	10	22	7.86
	Engineering	20	11	31	11.07

	Arts	38	25	63	22.50
Type of Institution	Federal University	122	85	207	73.93
	State University	16	8	24	8.57
	Private University	24	25	49	17.50

3.9.2 Availability of Mobile Devices

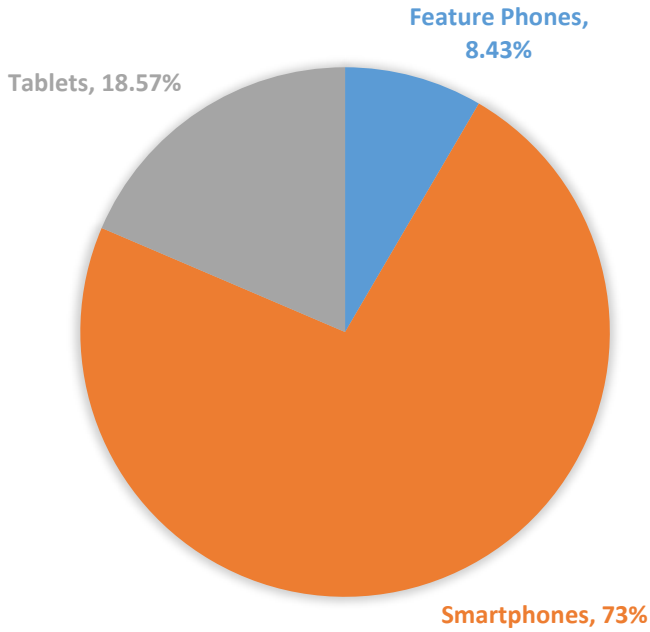


Figure 3.1: Students' Availability of Mobile Devices

Smartphones are ubiquitous and have found diverse uses in diverse fields. They put the power of a full-fledged computer in the average person's pocket. The survey revealed that 89.43% of the participants owned a phone. 81% of them had smartphones, while 8.43% had feature phones. The remainder (10.57%) of respondents had tablets.

Research question: Which device do you have?

Table 3.2: Students' Availability of Mobile Devices

Device Ownership	Respondents (N = 280)	Percentage (100%)
Smartphone	184	81.00
Tablet	64	10.57
Feature Phone	32	8.43

3.9.3 Mobile Operating Platform

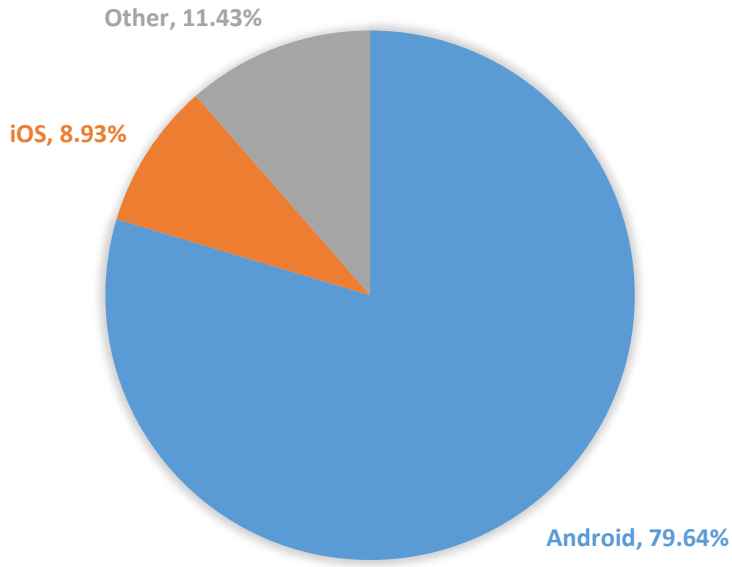


Figure 3.2: Students' Usage of Mobile Operating Systems

There are many mobile operating systems available in the market. They all share some basic functionality with different OSes in used being influenced by factors such as cost, ostentation and quality. This study revealed that the most popular mobile operating system by a wide margin is Android (83.64% of respondents). Apple's iOS is used by 4.93% of our respondents. The remainder of our respondents used other operating systems which include Blackberry, Windows Mobile, Symbian/Java Phones, or feature phones. These are encompassed by 'Other'.

Research question: Which mobile operating system you have in your mobile?

Table 3.3: Students' Usage of Mobile Operating Systems

Mobile Operating Systems	Respondents (N = 280)	Percentage (100%)
Android	209	83.64
iOS	39	4.93
Other	32	11.43

3.9.4 Internet Usage

Smart devices are heavily hampered when they have no Internet connection. Most of the useful features in smartphones are work using the Internet. This study (refer to Table 3.4) also demonstrated that 92.14% of participants use the Internet from their mobile devices

(including smartphones, tablets and normal (not smart) mobile phones). Only 15% of them do not have Internet connection in their mobile device.

Research question: Do you have regular Internet access on your mobile device?

Table 3.4: Students' Availability of Internet

Response	Respondents (N = 280)	Percentage (100%)
Yes	258	92.14
No	22	7.86

3.9.5 Purpose of using Internet

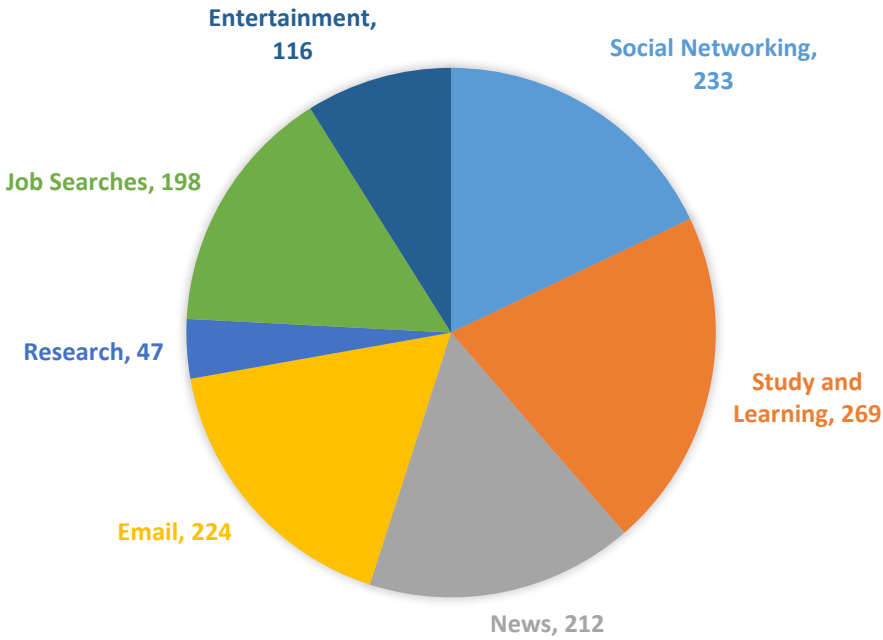


Figure 3.3: Participants' Purpose of Using Internet

The Internet holds a vast amount of information and people use it for many of their needs. I grouped purposes of using the Internet into seven broad categories and asked respondents to pick the answers that applied to them. The study revealed that the majority of people use the Internet for study (96.07%) and social networking (83.21%) purposes. 77.17% of people use the Internet for News, 75.71% for E-mail, 48.57% for Research, 16.79% for Job Searches, and 41.43% for Entertainment Purposes.

Research question: What do you use the internet for? (multiple answers permitted)

Table 3.5: Participants' Purpose of Using Internet

Purpose of Using Internet	Respondents (N = 280)	Percentage (100%)
Social Networking	233	83.21
Study and learning Purpose	269	96.07
News	212	77.14
Email	224	75.71
Research Purpose	47	48.57
Searching Job	198	16.79
Entertainment	116	41.43

3.9.6 Awareness of Mobile Learning Apps

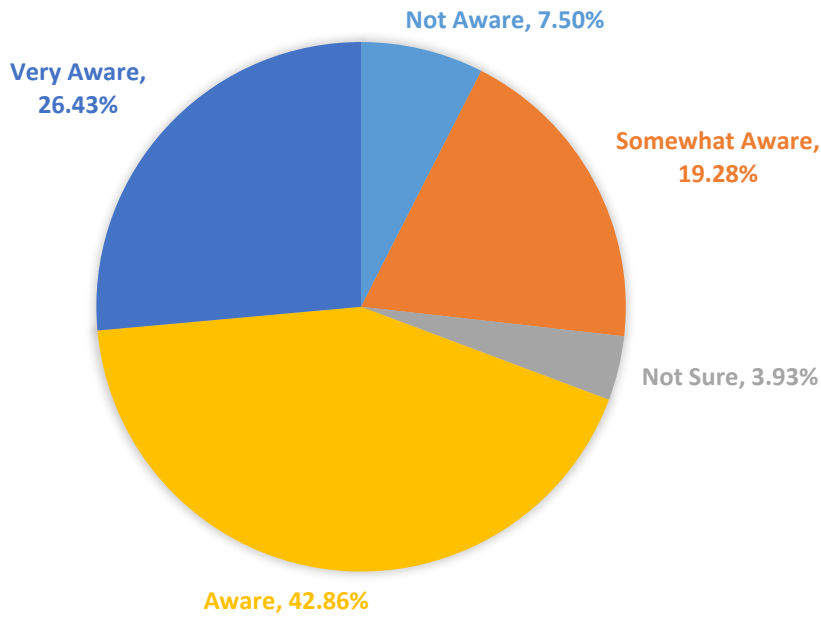


Figure 3.4: Students' Awareness of Mobile Learning Apps

This study explored the student's awareness of the various mobile learning apps (refer to Figure 3.4 or Table 3.6). The abbreviations used for this question are NA = Not Aware, SA = Somewhat Aware, NS = Not Sure, A = Aware, VA = Very Aware. The study shows that 69.28% students are aware of mobile learning apps (42.86% A and 26.43% VA).

Research question: Are you aware about mobile learning apps?

Table 3.6: Students' Awareness of Mobile Learning Apps

Awareness	Respondents (N = 280)	Percentage (100%)
Not Aware	21	7.50
Somewhat Aware	54	19.28
Not Sure	11	3.93
Aware	120	42.86
Very Aware	74	26.43

3.9.7 Duration of using Mobile Learning Apps

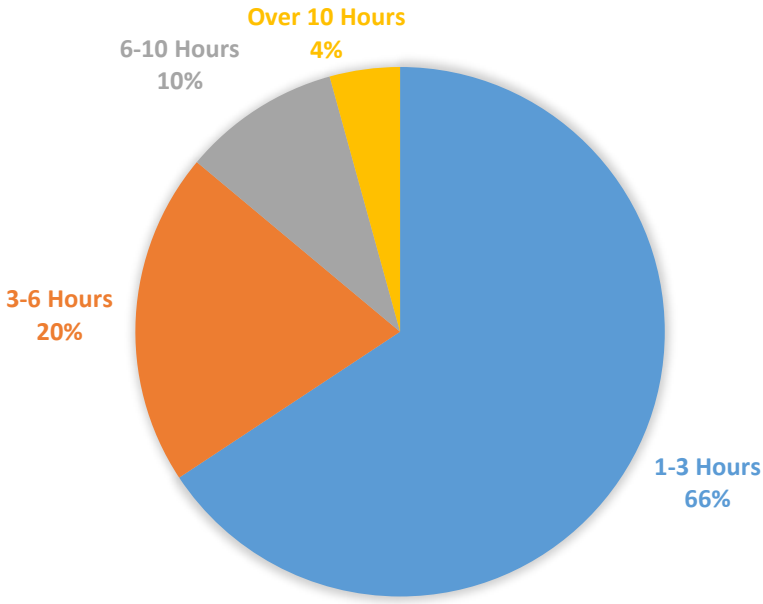


Figure 3.5: Students' Daily Usage of Mobile Learning Apps

Different people use different apps for different periods within the day. Regarding duration, the majority (65.14%) of respondents use mobile learning apps for 1-3 hours per day. 23.93% of them use for 3-6 hours per day; 9.64% use for 6-10 hours and only 4.29% use them for 10 or more hours per day.

Research question: How much time do you use mobile learning apps?

Table 3.7: Students' Daily Usage of Mobile Learning Apps

Duration	Respondents (N = 280)	Percentage (100%)
1 - 3 Hours	184	65.71

3 - 6 Hours	57	20.36
6 - 10 Hours	27	9.64
Over 10 Hours	12	4.29

3.9.8 Types of Mobile Learning Apps People Use

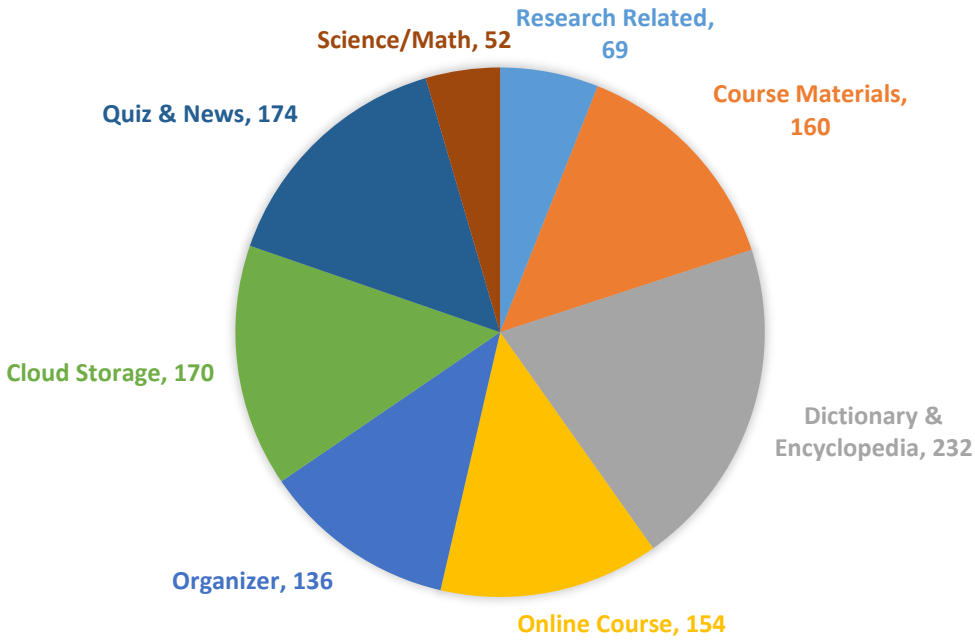


Figure 3.6: Students' Usage of Different Types of Mobile Learning Apps

Mobile learning apps are very popular among people for getting help in their studies and research as well as updating their knowledge. People use different kinds of apps for different purposes. Most of the respondents (82.86%) use dictionary and encyclopedia apps. Other categories of apps used by respondents are quiz and news apps (62.14%), cloud storage (60.71%), course materials and digital notes related apps (57.14%), language learning and taking online course apps (54.29%), organizer apps (48.57%), research related apps (24.64%), and science/math related apps (18.57%).

Research question: Which kind of mobile learning apps do you use? (Multiple answers permitted)

Table 3.8: Students' Usage of Different Types of Mobile Learning Apps

Type of Mobile Learning Apps	Respondents (N = 280)	Percentage (100%)
Research related apps	69	24.64

Course Materials	160	57.14
Dictionary And Encyclopedia	232	82.86
Language Learning/Online Course	154	54.29
Organizer/Other Useful Apps	136	48.57
Cloud Storage	170	60.71
Quiz and News	174	62.14
Science/Mathematics	52	18.57

3.9.9 Effectiveness of Mobile Learning Apps

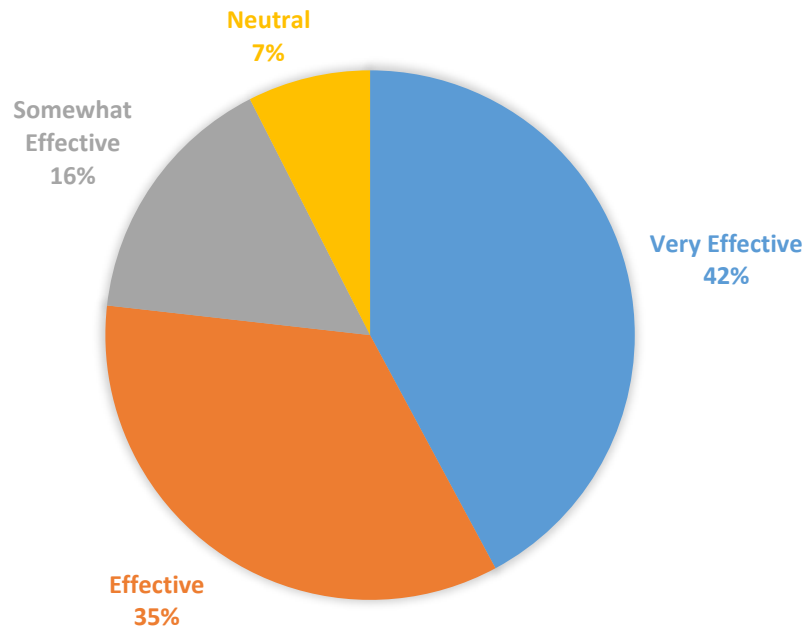


Figure 3.7: Students' Opinion about Effectiveness of Mobile Learning Apps

Mobile learning apps allow users to learn anytime and anywhere according to their convenience. They can easily be used to utilize free and leisure time. In terms of effectiveness, almost 42.14% of respondents find mobile learning apps 'very effective' whereas 34.64% of them find them 'effective'. 15.71% of respondents find the 'somewhat effective' and 7.50% of respondents had no clear idea and they were 'neutral'.

Research question: How effective do you find mobile learning apps?

Table 3.9: Students' Opinion about Effectiveness of Mobile Learning Apps

Measure of Effectiveness	Respondents (N = 280)	Percentage (100%)
Very Effective	118	42.14
Effective	97	34.64
Somewhat Effective	44	15.71
Neutral	21	7.50

3.9.10 Opinion On Impact of Mobile Learning Apps

There are millions of mobile apps available; many serving educational purposes. These apps help make the learning process run more smoothly and effectively. A large number of apps are freely available for download and can be used as the learner sees fit. I asked respondents about the integration of mobile learning apps in higher education and most of the respondents (87.86%) feel that mobile learning apps can make a great impact on higher education whereas 12.14% does not agree with this.

Research question: Do you feel mobile learning apps can have or are having a great impact on higher education?

Table 3.10: Students' View about Impact of Mobile Learning Apps over Higher Education

Response	Respondents (N = 280)	Percentage (100%)
Yes	246	87.86
No	34	12.14

3.10 Conclusion and Findings

In this chapter, I have outlined my quantitative methodology for collecting data and the insights that I wanted to uncover. The data says that people will favor a mobile app, hence the system will be mobile based with its data stored on the cloud. The following are the insights my data collection processes managed to glean;

- Potential users favor mobile applications. Developing a mobile solution meets these users on the platform with which they are most comfortable.
- By developing a cloud based solution, the educational content is stored in the cloud. This means that learning materials are accessible to the users no matter where they are in the world.
- Users are already comfortable doing several things including learning on their phones. This means that minimal change in user behavior is required for the system to reach adequate usage.

- Although many users have access to the internet, the cost of access still remains relatively high, especially for a developing country.

Chapter Four: System Design and Implementation

4.1 Preamble

This chapter outlines the methodologies and processes that I engaged to design and implement the system and its subsystems.

The working name of the application is '**GoLearn**'. In this chapter, I shall describe the system requirements, operating environment, system and subsystem architecture, files and database design, input formats, output layouts, human-machine interfaces, detailed design, processing logic, and external interfaces.

4.2 Analysis

4.2.1 SWOT Analysis

During the SWOT Analysis, I was able to identify some internal and external factors that were either positive or negative.

Strengths	**Weaknesses**
- Good knowledge of Android development - Good general knowledge of required skills, eg. UI, UX, Testing, Analytics - Experience with and access to essential development tools	- Lack of highly specialised skills in aspects asides programming - Low marketing expertise and budget
Oppotunities	**Threats**
- Good knowledge of Android development - Developing for a platform with massive reach. 1B+ users - Solution has features which cater to the economically challenged	- Better funded competitors may exert pressure - Poor electricity supply and hardware failures

Figure 4.1: SWOT (Strengths, Weaknesses, Opportunities and Threats) Analysis of My Project

4.2.2 Comparing 'GoLearn' to Other Solutions

During the course of my surveys, I was able to glean the features that were most important to students based on the data collected.

The features that students valued the most were:

1. Students did not spend a lot of time using these apps; therefore, it would be important to build a user interface that they could understand easily
2. Most students used their learning applications on mobile phones, therefore the app should be well suited to and optimized for smaller screens.
3. Although the majority of had basic internet access, most of them did not have the generous data plans that would allow them to constantly download large course materials of stream videos. I will speak more on this later.
4. More than 60% of students said that they used learning applications to view and manage their course content. Such an important feature had to be put into the app

4.3 Conceptualization of Features

4.3.1 Peer-to-peer sharing of course material

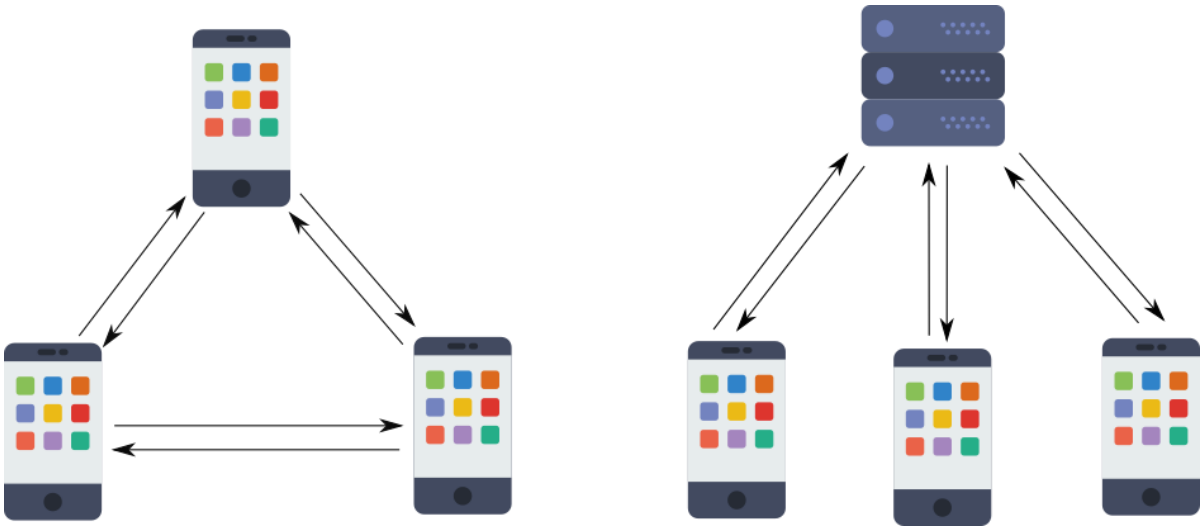


Figure 4.2: Simple models of a peer-to-peer network and a server-client network

Peer-to-peer (P2P) computing or networking is an architecture that partitions tasks or workloads between peers. Peers are equally privileged, equipotent participants in the application. They are said to form a peer-to-peer network of nodes.

Table 4.1: Client-Server vs Peer-to-Peer architecture

	Client-Server	Peer-to-Peer
Basic	There is a specific server and specific clients connected to the server.	Clients and server are not distinguished; each node act as client and server.
Service	The client request for service and server respond with the service.	Each node can request for services and can also provide the services.
Focus	Sharing the information.	Connectivity.
Data	The data is stored in a centralized server.	Each peer has its own data.
Server	When several clients request for the services simultaneously, a server can get bottlenecked.	As the services are provided by several servers distributed in the peer-to-peer system, a server in not bottlenecked.
Expense	The client-server are expensive to implement.	Peer-to-peer are less expensive to implement.
Stability	Client-Server is more stable and scalable.	Peer-to-Peer suffers if the number of peers increases in the system.

The Client-Server network model is very widely used. The server is a powerful system that data is stored upon. The client on the other hand, is the device which let the users access the data on the remote server.

The system administrator manages the data on the server. The client devices and the server are connected via a network (usually the Internet). This allows the clients to access data even if the client and server are physically far apart.

In a Client-Server model, the client process on the client device sends the request to the server process on the server machine. When the server receives the client request, it looks for the requested data and sends back a response.

As all the services are provided by a centralized server, there are chances of the server getting overloaded, reducing system efficiency. The server could get overloaded if it is not powerful enough or not employing good networking principles such as load balancing.

Unlike the Client-Server server model, the Peer-to-Peer does not distinguish between clients and servers. Instead each node can either be a client or a server depending on whether the node is requesting or providing the network services. Each node is considered as a peer.

To become a part of peer-to-peer network, a node must initially join the network. After joining it must start to provide services to and must request the services from other nodes in the peer-to-peer system. There are two ways to know which node provides which services;

they are as follow:

- When a node enters the peer-to-peer system, it must register the services it will be providing, into a centralized lookup service on the network. When a node desires for any specific service it must contact centralized lookup services to check out which node will provide the desired services. Rest of the communication is done by the desiring node and the service providing node.
- A node desiring for the specific services must broadcast the request for services to all other nodes in the peer-to-peer system. The node providing the requested service will respond to the node making the request.

Peer-to-Peer network has the advantage over client-server that the server is not bottlenecked as the services are provided by the several nodes distributed in a peer-to-peer system. On the other hand, if there are many nodes in a network and data needs to be send to nodes not directly connected to each other, transmission may be slow. This becomes more problematic as nodes in the network increase.

As I have said above, the peer to peer network allows nodes to communicate with one another equally. Some other benefits of peer to peer networks are;

1. P2P Network Nodes Can Communicate Without a Data Connection
2. P2P Networks Generally Have Better Throughput When Transmitting Data

4.3.1.1 Communication Without a Data Connection

P2P network nodes can communicate without a data connection. This is especially useful for places in which data costs are prohibitively high. In Nigeria and other developing countries data costs are either very high or most people aren't economically prosperous enough to afford them. This poses a problem for people looking to share educational and; earning materials over the traditional internet as it would incur data costs on the users. A simple solution to this can be found via P2P networks. With wireless p2p networking technologies such as Wi-Fi direct and others implemented in mobile devices, a host device can create a wireless hotspot through which devices can connect and download files. Such a local connection will not touch the broader Internet and as a result no data costs will be incurred. Only that of downloading the software in the first place if it is distributed via an online marketplace such as the windows app store, apple app store or Google play store.

In rural areas where internet connections are slow, expensive or non-existent, such solutions allow students to have access to their course materials as soon as one person within close proximity to them does. This gives students no reason to not have the course material especially given the economic situation that we have in Nigeria.

4.3.1.2 Better Throughput When Transmitting Data

P2P networks (that are relatively small) generally have better throughput when transmitting data. They perform closer to the ideal speeds of the network standards since they do not have to deal with things such as transmitting over the whole Internet. Since these p2p networks for wireless hyper local networks, they do not need to go thought the broader internet for data transmission. For this reason, these p2p networks offer higher throughput especially when

the number of peers are kept low.

In places with high network congestion,

4.3.2 Cloud Timetable

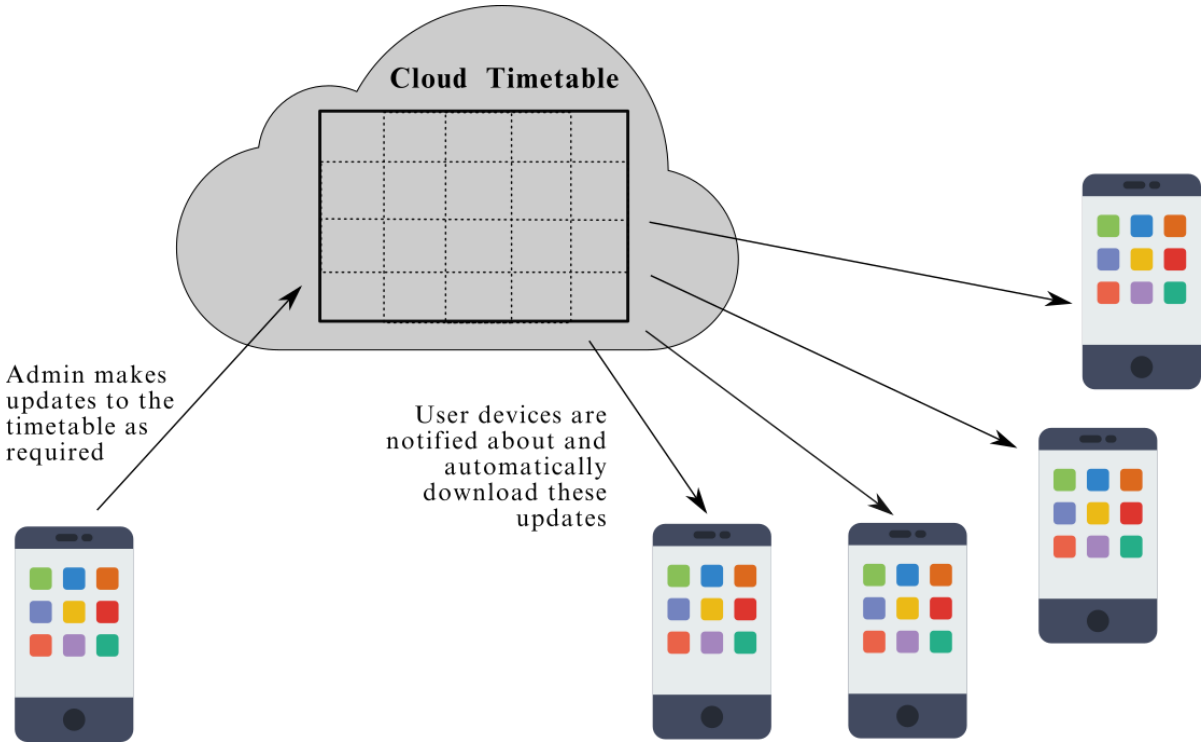


Figure 4.3: A diagrammatic representation of the Cloud Timetable

The concept of a cloud timetable is that we have a timetable which displays all the courses that the class will have and when they shall be holding. If there are any changes made to this timetable by the administrator, then all member of the class that are subscribed to this timetable will be informed either via email, text, mobile notification or some other means.

This made especial sense for the University as course schedules aren't always set in stone and change from time to time. I have experienced this several times during my stay at the University. There have also been several cases where students went very late to or completely missed exams because they didn't know that the time or venue of the paper had been changed earlier. A cloud timetable can help to fix this.

A great deal of this information is passed along over WhatsApp and students without a smartphone or mobile data might suffer as a result of this. It is for this reason that I listed text messages as one of the reasons for communicating these changes to students.

4.4 Development Methodology

4.4.1 Agile Method

For this project, I went with the Agile Development Methodology. Agile software development is an approach to software development under which requirements and solutions evolve through the collaborative effort of self-organizing and cross-functional teams and their customer(s)/end user(s). It advocates adaptive planning, evolutionary development, early delivery, and continual improvement, and it encourages rapid and flexible response to change. There is significant anecdotal evidence that adopting agile practices and values improves the agility of software professionals, teams and organizations

One of the reasons for this is that given the nature of the final year project, I would need to rapidly piggy back with my supervisor and potential users and stakeholders while developing the system. The Agile method emphasizes this iterative and rapid development.

Every few weeks, I had code sprints which saw trying to churn out working code.

Below are the principles of agile development and how I followed them.

4.4.1.1 Individuals and Interactions over processes and tools

If processes and tools are seen as the way to manage product development and everything associated with it, people and the way they approach the work must conform to the processes and tools. Conformity makes it hard to accommodate new ideas, new requirements, and new thinking. Agile approaches, however, value people over process. This emphasis on individuals and teams puts the focus on people and their energy, innovation, and ability to solve problems. You use processes and tools in agile project management, but they're intentionally streamlined and directly support product creation. The more robust a process or tool, the more you spend on its care and feeding and the more you defer to it. With people front and center, however, the result is a leap in productivity. An agile environment is human-centric and participatory and can be readily adapted to new ideas and innovations.

I decide to value individuals and interactions highly, and realized the following benefits:

1. Communication is clear and effective.
2. Communication is quick and efficient.
3. There are more chances to innovate.
4. Processes can be customized as necessary.

And I avoided the following problems:

1. Becoming overly reliant on processes instead of finding the best ways to create good products.
2. One process doesn't fit all projects or stakeholders.

3. Ambiguous and time-consuming communication between stakeholders.

4.4.1.2 Working Software over comprehensive documentation

A software development project's focus should be on producing working products. The second Agile core value emphasizes working software over comprehensive documentation.

On projects using agile management tools, the only way to measure whether you are truly done with a product requirement is to produce the working product feature associated with that requirement. For software products, working software means the software meets what's called the definition of done: at the very least, developed, tested, integrated, and documented. After all, the working product is the reason for the project.

On a traditional project, if you're 75 percent done, you don't have any working software to release to users. "75 percent done" traditionally means you're 75 percent in progress and 0 percent done. On an agile project, however, if you're 75 percent done, you have working product features for 75 percent of your project requirements — the highest-priority 75 percent of requirements.

Although I documented my project, my major focus was making my software work and making it work well for my users. After a sufficient level of development had been done, I began my documentation. This saved me time and resources as I did not need to document features that did not make it into the final system implementation

4.4.1.3 Customer Collaboration over contract negotiation

Given that this is a school research project, I did not have any customers per se and as a result no contracts to negotiate. All the same, I had users of my system. I took care to take my users along and tell them about the features I was developing. I also took in their input on upcoming features and their feedback and criticism on already implemented ones.

4.4.1.4 Responding to Change over following a plan

Change is a valuable tool for creating great software products. Using agile management principles, I was able to respond to users and changes in their requirements at lightning pace. This gave me an upper hand as I was able to develop a relevant and helpful product that people want to use.

Unfortunately, traditional project management approaches attempt to wrestle the change monster to the ground and pin it down so it goes out for the count. I took serious care not to do this. I instead embraced changes to mean that users felt my system was useful and wanted it to better address their needs. Traditional project teams often find themselves blindly following a plan and missing opportunities to create more valuable products. By contrast, agile projects accommodate change systematically. The agile approaches to planning, working, and prioritization allowed me to respond quickly to change. The flexibility of these agile approaches actually increased project stability, as change on agile projects is predictable and manageable.

As new events unfolded, I was able to incorporate those realities into the ongoing work. For this project, new items were an opportunity to provide additional value instead of an obstacle

to avoid. This gave me a greater opportunity for success.

4.5 System Requirements Engineering

For any project to be successful, its requirements must be clearly thought out, balanced and understood as well as based on user needs. This step is very important because having good requirements will allow the system to be developed properly and within the time constraints. Project creep is also avoided. From, the previous chapter, Methodology, through data collection, we established the features that are most important to the users of the system. They are;

- Ability to access (download and share) organized and up-to-date course content
- A simple and pleasant user interface
- An easily accessible, up-to-date and customizable timetable
- Ability to work decently offline
- An app they could access easily, hence we chose mobile

4.5.1 Types of Requirements

The system has both functional and non-functional user requirements

4.5.1.1 Functional Requirements

Typical functional requirements are;

1. Business Rules
2. Transaction creation, correction, adjustments and cancellations
3. Administrative functions
4. Authentication
5. Authorization Levels
6. Audit Tracking
7. External Interfaces
8. Certification Requirements
9. Reporting Requirements
10. Historical Data
11. Legal and Regulatory Requirements

For The functional requirements of the system are as follows;

1. **The users should be able to sign in to the application, give a few details about themselves, their university and current class and then be able to access course materials and the class timetable easily.**
2. **The admin users should be able to add new course content and upload new course material to the cloud storage.**
3. **The admin should be able to add to, edit and delete the contents of the timetable.**
4. **Users should be able to easily share course material without an internet connection; this uses P2P networking.**
5. **The user should be able to change details about themselves such as current level. This is useful for when a new school session starts,**

4.5.1.2 Non-Functional Requirements

Typical non-functional requirements include;

1. Scalability
2. Availability
3. Reliability
4. Recoverability
5. Maintainability
6. Serviceability
7. Security
8. Manageability
9. Data Integrity
10. Usability
11. Interoperability

The non-functional requirements of the system are as follows;

1. **All components of the system (the mobile application as well as the backend) must respond to users quickly**
2. **The app must be scalable**
3. **The UI must be easily understood and used**
4. **The timetable must stay updated as long as the user's device is connected to the internet or another user (via a P2P network)**
5. **The course must stay updated as long as the user's device is connected to the internet or another user (via a P2P network)**
6. **The Viewers for course materials (PDF, Word Document, images, audio and video)**

must display the materials properly

7. **The upload and download speeds must be good**

4.6 Hardware and Software Requirements

Table 4.2: Hardware and Software Requirements for the system and the development computer

	Hardware Requirements	Software Requirements
Linux/Windows PC	**RAM:** 4GB (8GB Recommended)	**OS:** Ubuntu 16.04/Windows 10
	Disk Space: 20GB	**Dev Tools:** Android Studio Version 3.0 (Version 3.4 Recommended)
	Processor: 1.8GHz Dual Core (2GHz Recommended)	
Android Smartphone	**RAM:** 1GB (2GB Recommended)	**OS:** Android Jelly Bean 4.1.x, API 16 (Android KitKat 4.4, API 19 Recommended)
	Disk Space: 512MB	**On Device API:** Google Play Services
	Processor: 1GHz Dual Core	
Ubuntu Server	**RAM:** 8GB RAM (16GB Recommended)	**OS:** Ubuntu Server 16.04

	Disk Space: 10GB SSD Storage	**Runtime:** Node.js v8
	Processor: 2.5GHz Dual Core (3.4GHz Recommended)	

4.7 System Overview

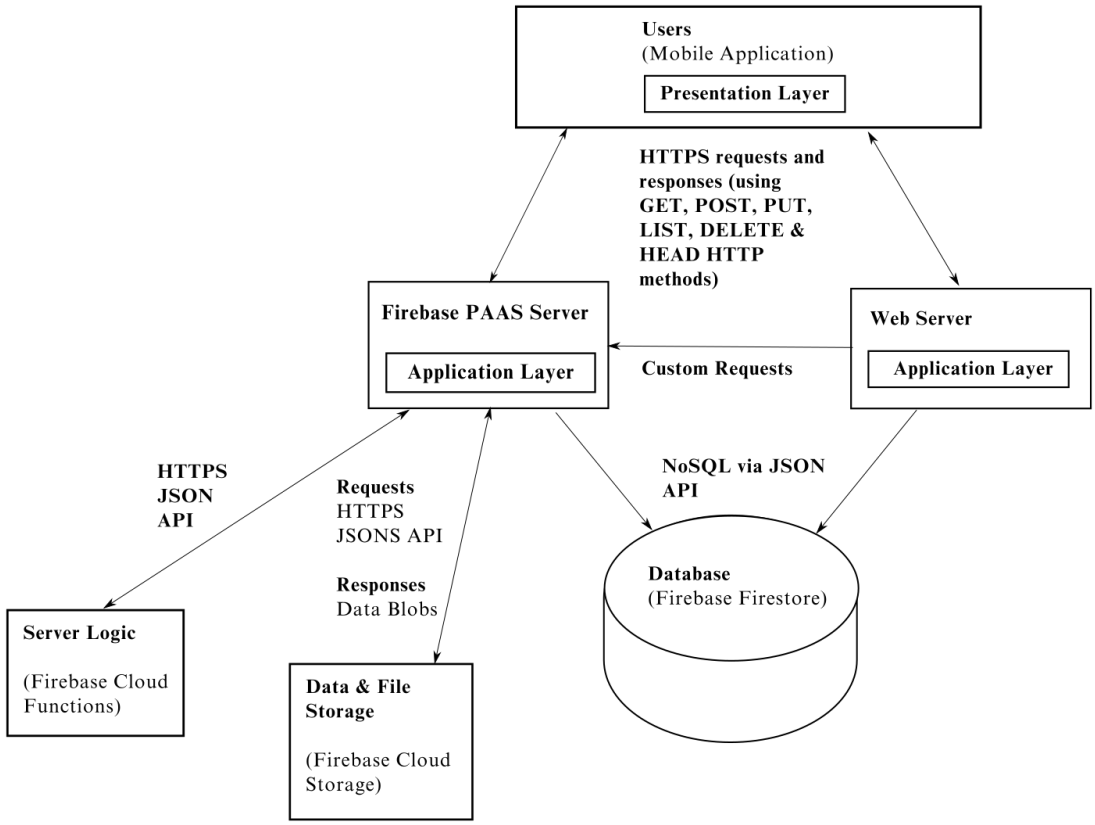


Figure 4.4: System Architecture Diagram

For this application the users (students) will have a mobile application. With this application, the students will be able to upload and download course material to and from a cloud storage server. The system follows the three tier application system.

4.7.1 Presentation Layer

This consists of the mobile application. This is an android app built using the traditional Android toolkit (i.e. Android Studio as the IDE, Java/Kotlin as the Programming Languages and XML for the UI Design)

4.7.2 Application Layer

This consists contains the following

1. **The Firebase Server**
2. **Personal Web Server**

4.7.3 Data Layer

This consists the following

1. **Data and File Storage Server**
2. **Database**

This section describes the system in narrative form using non-technical terms. It should provide a high-level system architecture diagram showing a subsystem breakout of the system, if applicable. The high-level system architecture or subsystem diagrams should, if applicable, show interfaces to external systems. Supply a high-level context diagram for the system and subsystems, if applicable. Refer to the requirements trace ability matrix (RTM) in the Functional Requirements Document (FRD), to identify the allocation of the functional requirements into this design document.

4.8 Design Constraints

4.8.1 Software Constraints

4.8.1.1 Platform Constraints

The main constraint encountered during development was that posed by the nature of the iOS networking APIs. I designed a prototype using the React-Native framework and quickly realized that iOS has much more stringent rules for connecting to Wi-Fi networks programmatically. This made it impossible to implement the P2P file sharing feature in a user friendly manner. The reason that it would not be user friendly is that to access;

Table 4.3: Table outlining the differences between the P2P sending process on iOS and Android.

Step	iOS	Android
1.	The user has to **manually** navigate to the	-

	Settings app.	
2.	The user has to **manually** disconnect from any wireless networks the device is currently connected to.	Disconnecting from other wireless networks can be done **programmatically**.
3.	The user has to **manually** connect to the hotspot generated by the device being used as a server in the P2P network (hotspot device).	Finding and connecting to the hotspot generated by the hotspot device can be accomplished **programmatically**.
4.	On both Android and iOS (the more recent OS versions), hotspots must have alphanumeric passwords of at least 8 characters since they adhere to the WPA2 standard. The process of **manually** entering a password such as "A2jnw$2jd" is error prone and wastes time.	Scanning a QR Code that encodes the details of the hotspot (SSID and password) with the camera of the client device allows it connect to the hotspot device **programmatically**.

In addition to the above reasons, iOS makes a minority of the devices used by students on campus, (generally below 10%). Focusing time and resources on the Android app allows me to develop a better app for the majority of the vast majority of potential users.

I therefore decided to develop my device for only android.

4.8.2 Backend Constraints

Part of being a good systems developer involves deciding when to write your own code and when to use an external using library or API. Most of the system's backend (application and data tiers) are developed using Firebase. I will outline the reasons for this using some of the features/services provided by Firebase.

1. **Firebase Auth** - This allow use to manage the user's register/sign in flow using Firebase tools. In addition to allowing me to focus my efforts on more functional system requirements, this service allows me to to let users to authenticate themselves via external services such as Google, Facebook, Twitter, or their phone numbers.
2. **Firebase Cloud Storage** - This allows user data (course material such as PDFs, Office Documents, Videos, Images and Audio) to be stored on the Firebase servers. This provides me with and scalable and secure data storage server and an easy-to-use API for the uploading and downloading of data.
3. **Firebase Firestore (Database)** - This provides a highly available and scalable online NoSQL database to store user data and metadata as well as course metadata. The API provides secure access control and data query features.
4. **Firebase Cloud Functions** - This provides a BAAS (Backend-as-A-Service). It precludes me from having to manage servers and allows me to execute system functionality based on events that occur in the system such as writing to a database,

querying from a database, uploading or downloading a file and a normal HTTP/HTTPS request.

5. **Firebase Cloud Messaging** - This allows me to send messages to all devices with my application installed. This means that I can notify all devices signed in to a particular University course and level when new content has been added or old content has been updated to (optional) prompt the user to download the latest version of the material. This also allows the timetable to be updated on every device immediately a change is made and confirmed by the admin.

For the above reasons, I implemented a great deal of my backend functionality with Firebase.

4.9 Future Contingencies

In the event that Apple loosens the aforementioned restrictions on the iOS networking APIs or iOS devices begin to consist a more significant proportion of the targeted users, an iOS version of the app may become pertinent.

4.10 File and Database Design

4.10.1 Local File Design

The files in the project are the course materials. They are stored in a folder named 'GoLearn'. The hierarchy is as follows;

1. The 'GoLearn' folder is the root folder
2. Inside the 'GoLearn' folder is a 'Courses' folder
3. Inside the 'Courses' folder are the folders containing Course Materials. They are in the format: '**[course-code]** - **[course-title]**'. For a course like CSC 409 the folder will be named; 'CSC 409 - Data Communications and Networks'.
4. Inside the course folders are exactly two folders; 'Documents' and 'Videos'
5. Inside the 'Documents' folder is course material such as PDFs, Office Documents and text files.
6. Inside the 'Videos' folder are video materials and links to YouTube Videos.

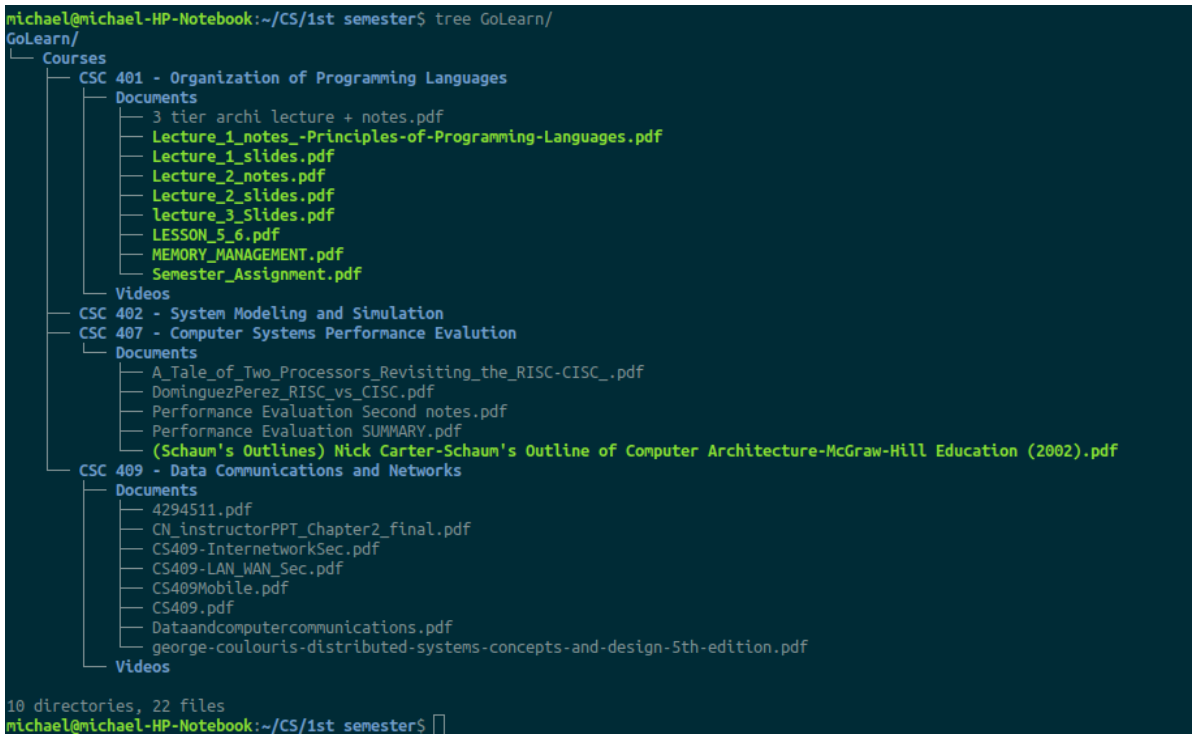


Figure 4.5: Application Folder structure

4.10.2 Cloud Storage (Remote File) Design

Before course material is uploaded to the cloud, it is compressed using the GZIP standard. The reason for this is three fold;

1. To reduce bandwidth costs for the admin while uploading and the students while downloading course material
2. To reduce upload and download times
3. To reduce cloud storage costs for the system

The compressed file is then uploaded to the cloud, and the file metadata is stored in the database.

When a student wants to download a course material, the application sends a request to the cloud storage server which serves the file to the app. The compressed file which is stored in the cloud then begins downloading. After the download is complete the file is decompressed to its original format and stored on the student's local storage. From it can be viewed normally.

4.10.3 Database Design

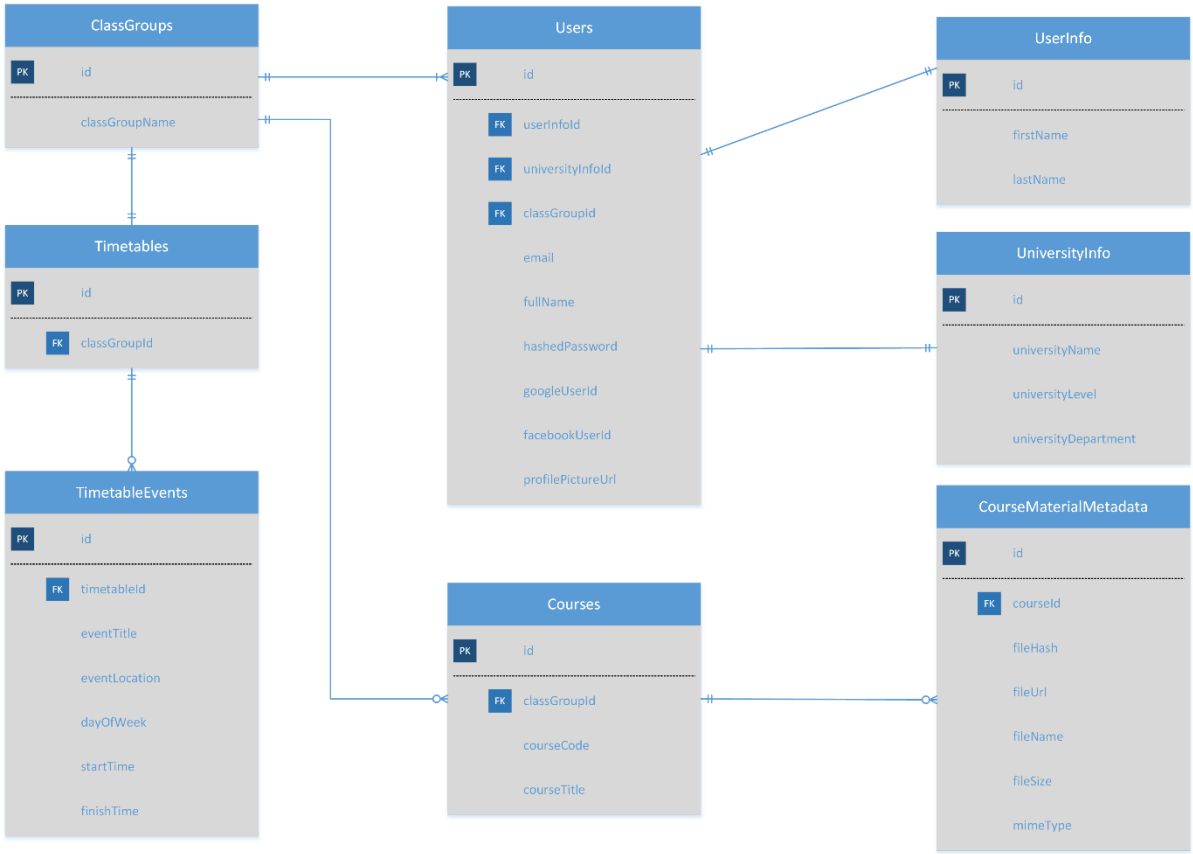


Figure 4.6: Entity Relationship Diagram (ERD) for the system's database

The system makes use of a NoSQL database. One of the reasons that I went with a NoSQL database is the extra flexibility doing so grants. In line with my Agile Methodology, I was freed from having to know exactly how my data would look at the beginning. On the other hand, using a schema-less NoSQL database as I did can create the concern of bad data organization. I prevented this problem by having keeping track of all the possible fields in each document.

I had the following collections in my NoSQL Firestore database.

1. **User** – This collection holds the basic user info such as email, hashed password, username etc.
2. **UserInfo** – This collection holds extra information about users asides from authentication details. It is used to hold information such as first name, last name, age/date of birth, and user settings.
3. **UniversityInfo** – This collection holds information about the user's University and course of study. It holds information such as name of University, Department, and level.
4. **Courses** – Each document in this collection, holds the courses that have been added by the admins of each **group.**
5. **CourseMaterialMetadata** – This holds metadata about the files that have been uploaded to the cloud storage. It holds information such as the original name. File type,

file size, compressed size, file hash and more.

6. **ClassGroups** – This holds data about the groups that have been created in the app. It holds a reference to the members of the group and the admin(s).
7. **Timetable** – This holds the data about the courses that the students will be having and when.
8. **TimetableEvents –** This holds the details about an event such as a lecture.

4.11 Use Cases

4.11.1 Students

- The students use the system to view their courses and course material.
- The students use the system to view their timetable.
- The students use the system to share course materials via P2P networking.
- The students register/sign in to the system

4.11.2 Admin

- The admin uses the system to add/edit/delete courses and course material
- The admin uses the system to add/edit/delete entries on the timetable.
- The admin uses the system to send important updates to the students
- The admin registers/signs in to the system

4.11.3 Use Case Diagram

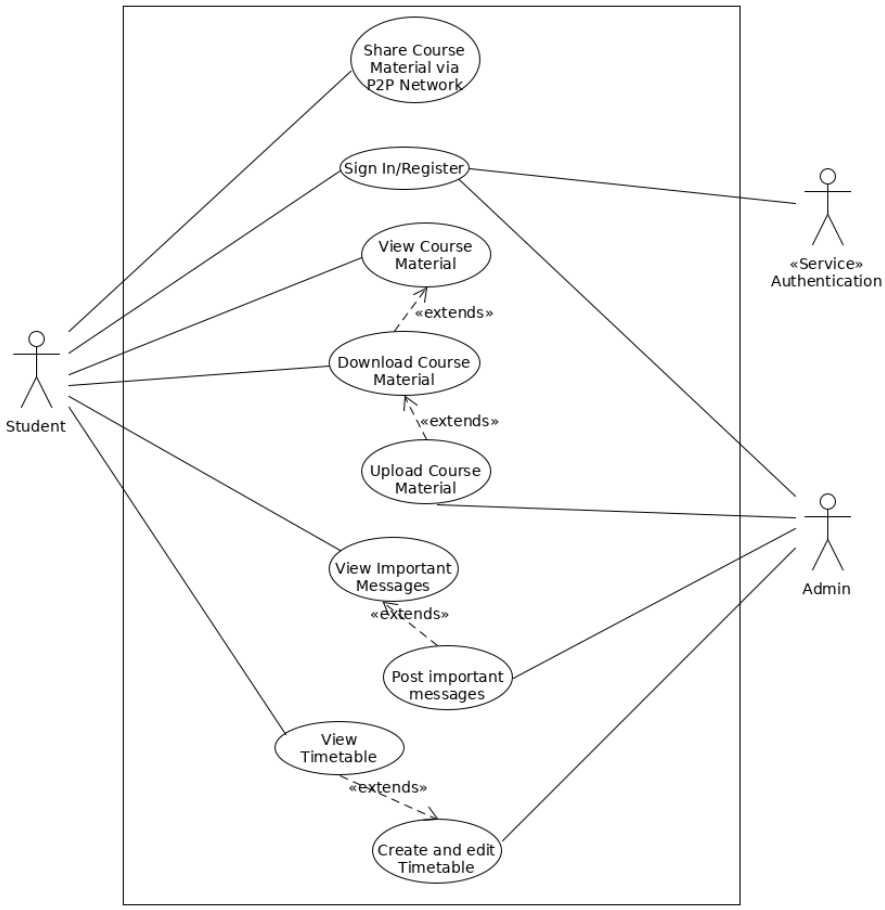


Figure 4.7: Use Case Diagram of My Project

4.12 Visual Design

The user interface was designed with XML as is usually the case for android.

4.12.1 Application Icon

Figure 4.8: This gives and illustration of the process of designing the application icon

A visually descriptive and well-designed icon is always important in making a compelling application. When designing the icon, it was important to have a simple, clear and clean design that would reflect the application's values such educational empowerment and ease of use.

4.12.2 Description of the Icon's components

- **The Board** - The board is one of the most important tools in a learning environment. In this context, I'm taking board to also mean a surface on which slides can be projected.
- **The 'g'** - The 'g' stands for 'GoLearn' which is the working title of the application. The blue and green symbolize serenity and calmness (two conditions that are important to the learning process). The use of gradients in the color scheme (the gradual transition from blue to green) says that the app is flexible to the users' needs and will be improved and updates as requirements change.
- **The Complete Icon** - This symbolizes that all the qualities of the 'g' are being brought to the learning environment.

4.13 Application Components

Splash Screen

Figure 4.9: The application's splash screen

The Splash screen improves UX by showing the user a relevant image instead of a plain black or white screen when the app is being loaded. It helps prime the user for their coming experience.

The Sign in / Register Screen

The sign in/register flow allows users to authenticate themselves and get access to online material for their specific group. The authentication is done with an email address and a corresponding password. The password must not be less than 8 characters.

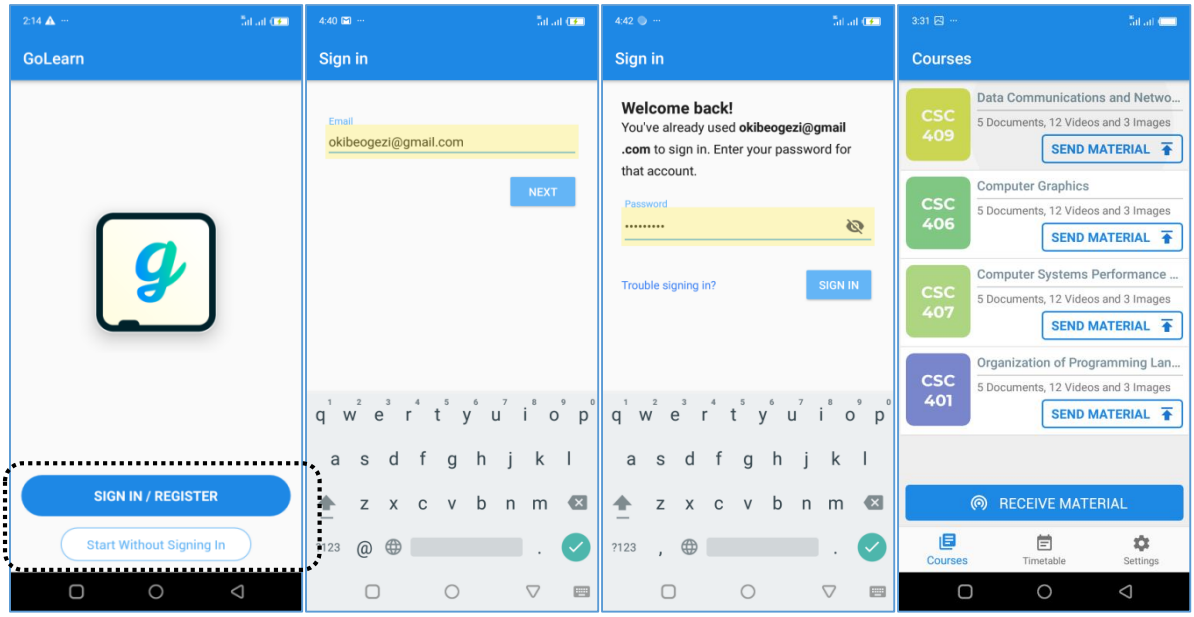


Figure 4.10: The Application's Sign in/Register Screen

This screen has two buttons namely;

i. Sign in/Register Button
ii. Start Without Signing in Button

The Sign in/Register Button

When this is pressed, the authentication flow begins.

Activity Diagram

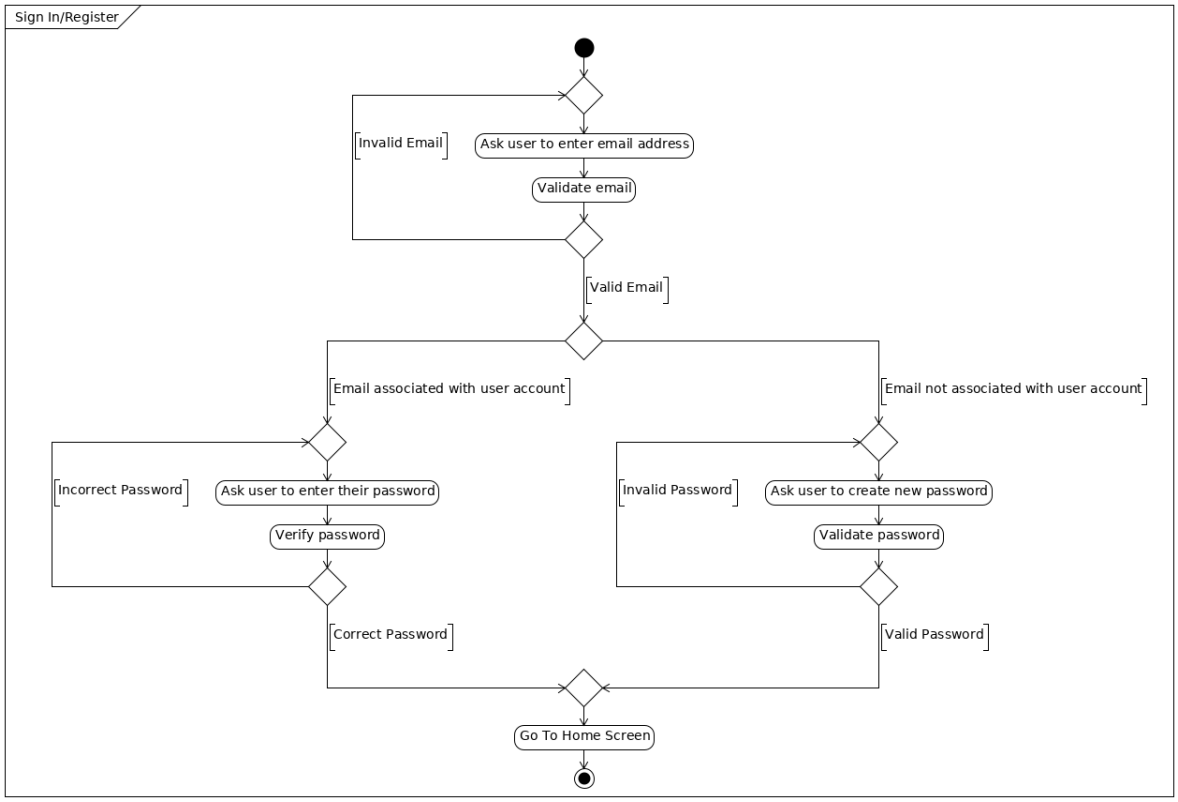


Figure 4.11: Activity Diagram for Sign in/Register

The Start Without Signing in Button

When this is pressed, the user is allowed to use the app immediately (goes to the home screen), but will lack features such as the ability to be download course content from the cloud and the online timetable.

The Home Screen

The home screen is the first screen an authenticated user sees. This screen contains three tabs mounted on a bottom navigation panel. The reason for the bottom navigation instead of something like a hamburger menu is that I wanted users to be about to easily move between the root view. Ample UI research has shown that bottom navigation menus are the best was to do this. Also both Apple and Google recommend this design pattern.

The three root views on the home screen are a follows;

1. **Courses** - This shows the list of courses that are available to the user.
2. **Timetable** - This shows the most recent timetable for the courses that the user is taking.
3. **Settings** - This shows the user's settings. The admin can add new courses from here. Users can also change some of the information that they have provided or sign out.

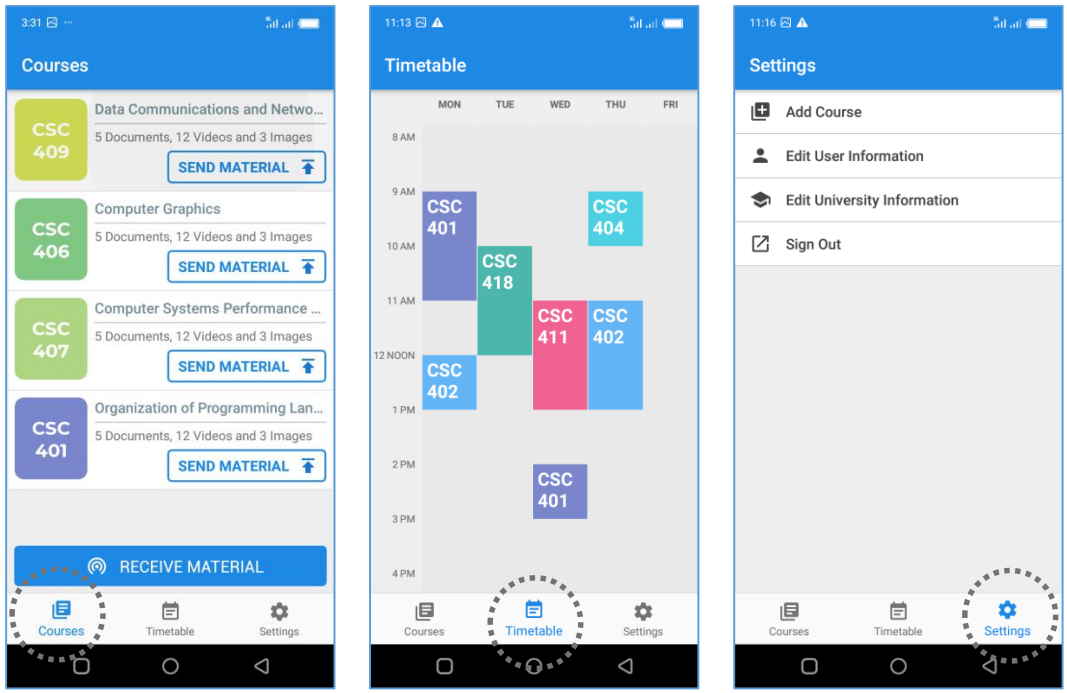


Figure 4.12: This shows the home screen with the three root views; Courses, Timetable and Settings

The Add Course Screen

This allows the admin to add courses to the system for the class group in question. The admin needs to provide correct course code and course title details. After these entries are validated, the new course is created on the database and the admin is prompted to ass new material to it.

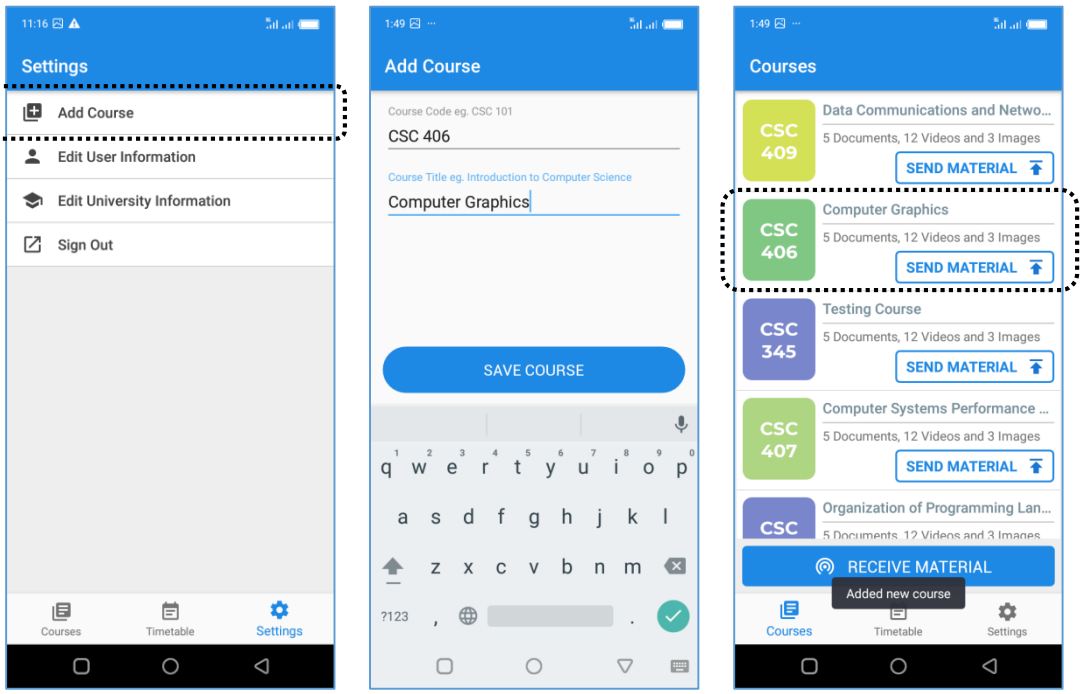


Figure 4.13: The flow of adding a course using the app

Activity Diagram

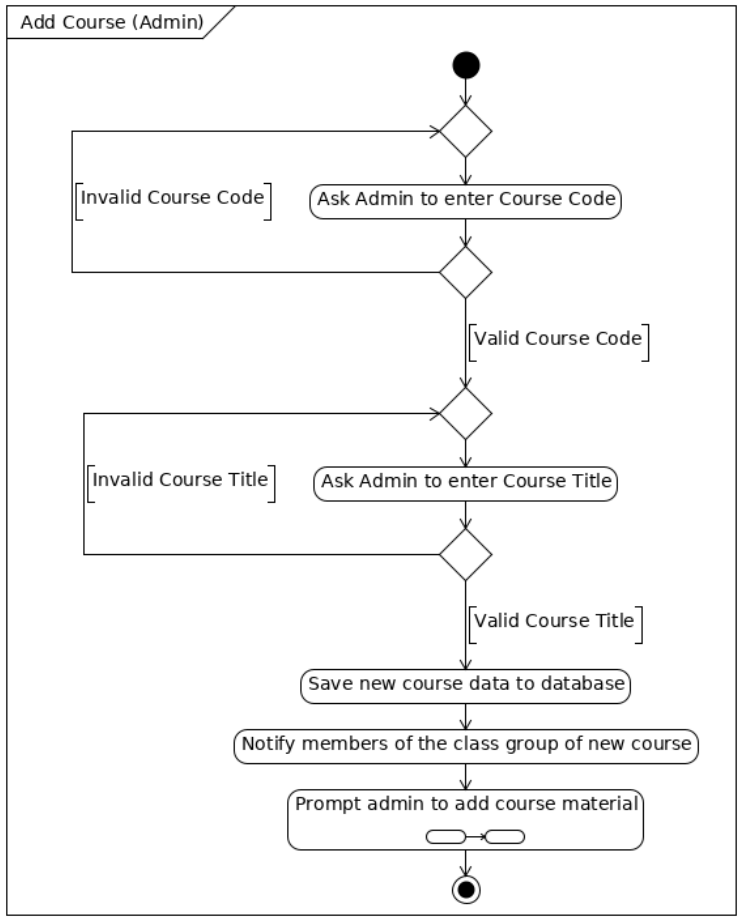


Figure 4.14: Activity Diagram for Adding Course

The Add Course Material

Here the admin chooses a file that they want to add to a course as one of tits course materials. After the file is picked and a suitable title is given to it, the admin can save the material. At this point the material is compressed using the GZIP standard and uploaded to the system's cloud storage.

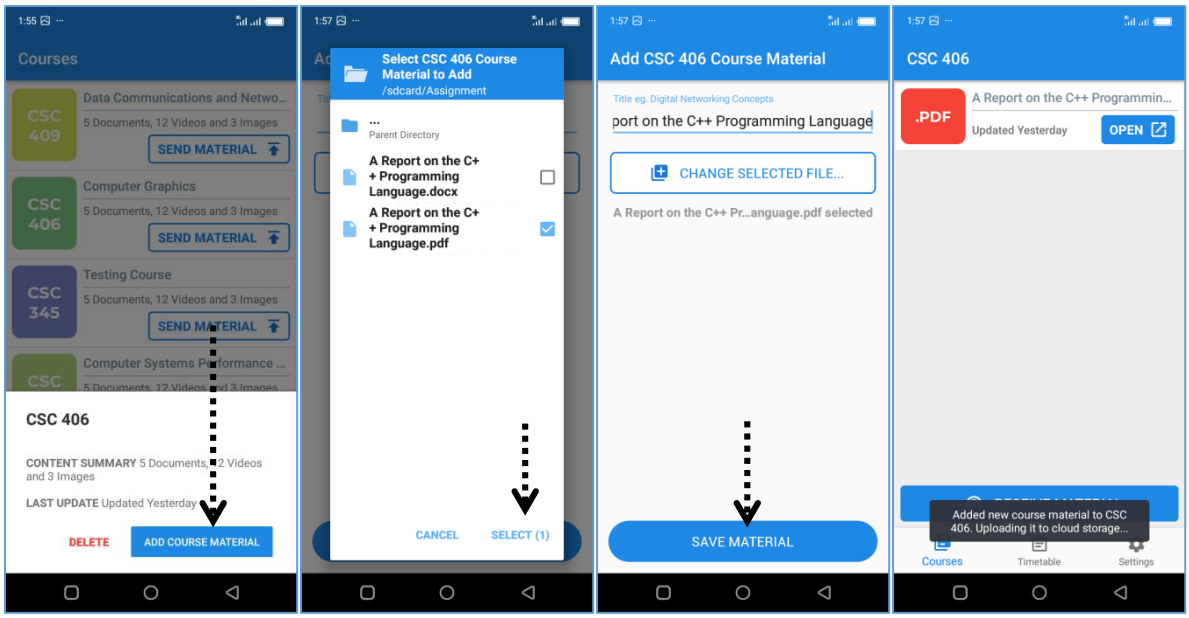


Figure 4.15: The steps in adding course material a course using the app

1. The first screenshot shows, the menu that appears after long pressing one of the course at the root view of the app. We see options to ‘Delete’ or ‘Add Course Material’. We press the latter and proceed to the next screen.
2. We tap the ‘Select File...’ button and a file picker dialog is launched. This allows a file to be chosen. The chosen file can be changed multiple times, before advancing.
3. After the file is selected and a name is picked, we tap on ‘Save Material’. This saves the file to the apps folder and begins the upload process.
4. We are now taken to the view of documents and materials for the course we just added the material to. We are notified about the upload process too.

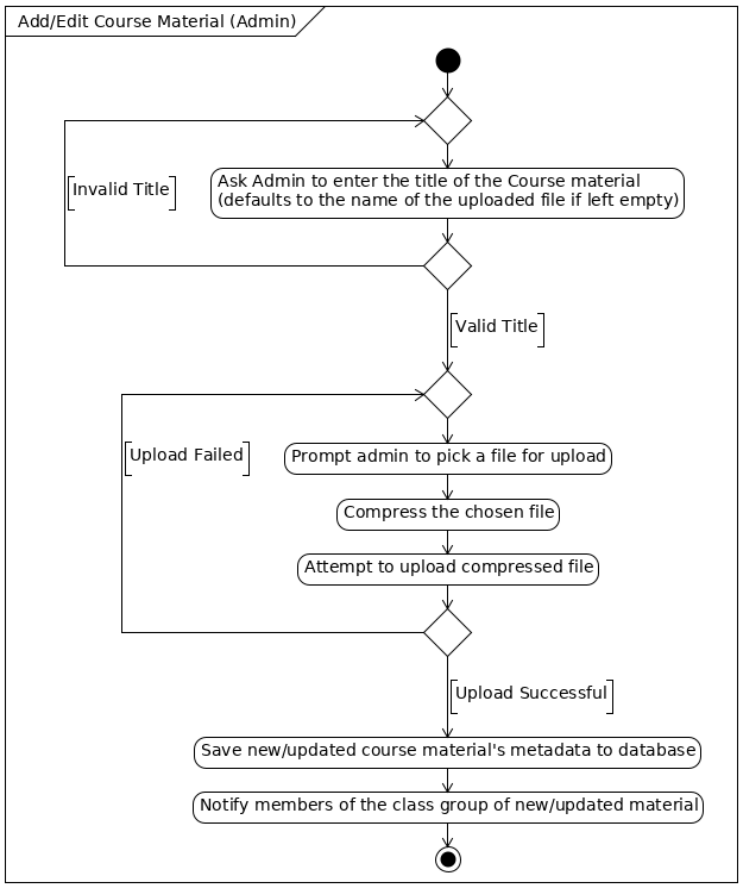


Figure 4.16: Activity Diagram for Adding Course Material

Course Material Viewer

The ‘Courses’ tab on the app’s home screen shows a list of the courses that the student has. Tapping on any of these courses opens a list of the course materials that are part of this course. Tapping on any course material on this list, opens it in a viewer. If no viewer on the device can open the material, then the user is directed go download one from the app store.

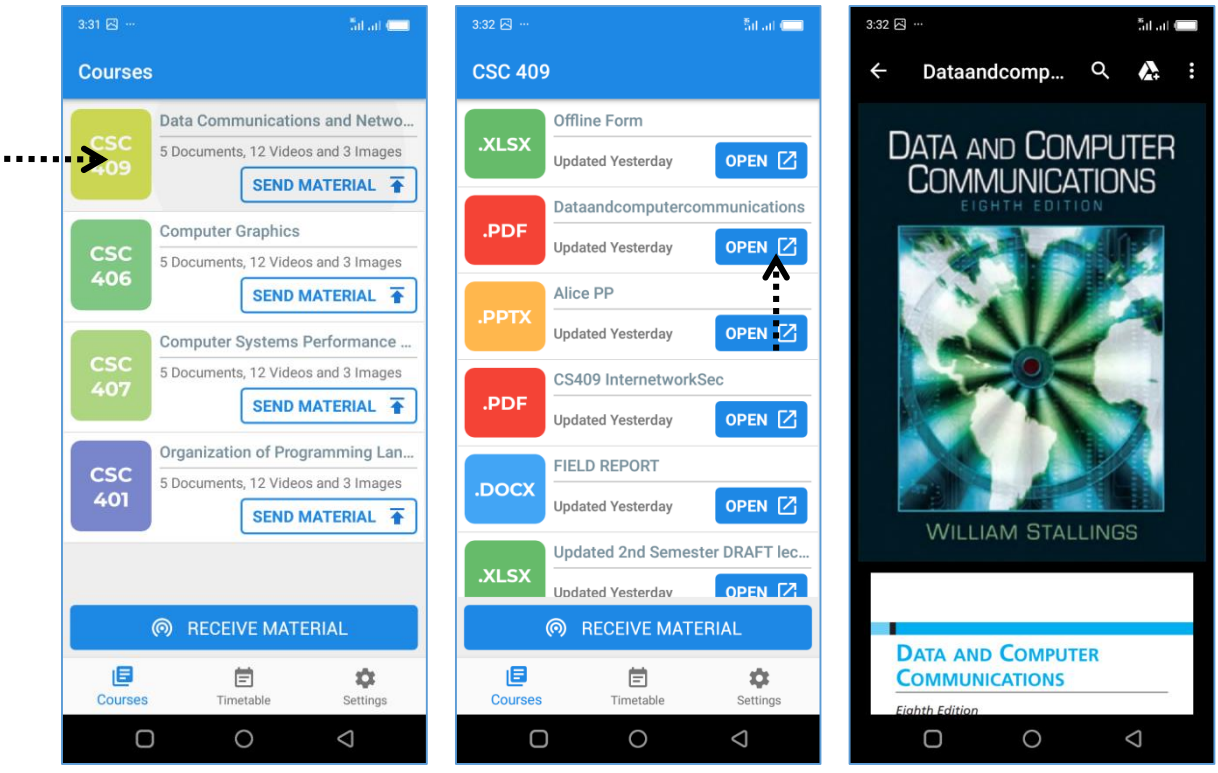


Figure 4.17: The process of opening course material

Timetable

The timetable feature allows students to have a skeuomorphic and up-to-date timetable for their courses.

Add/Edit/Delete Timetable Entry

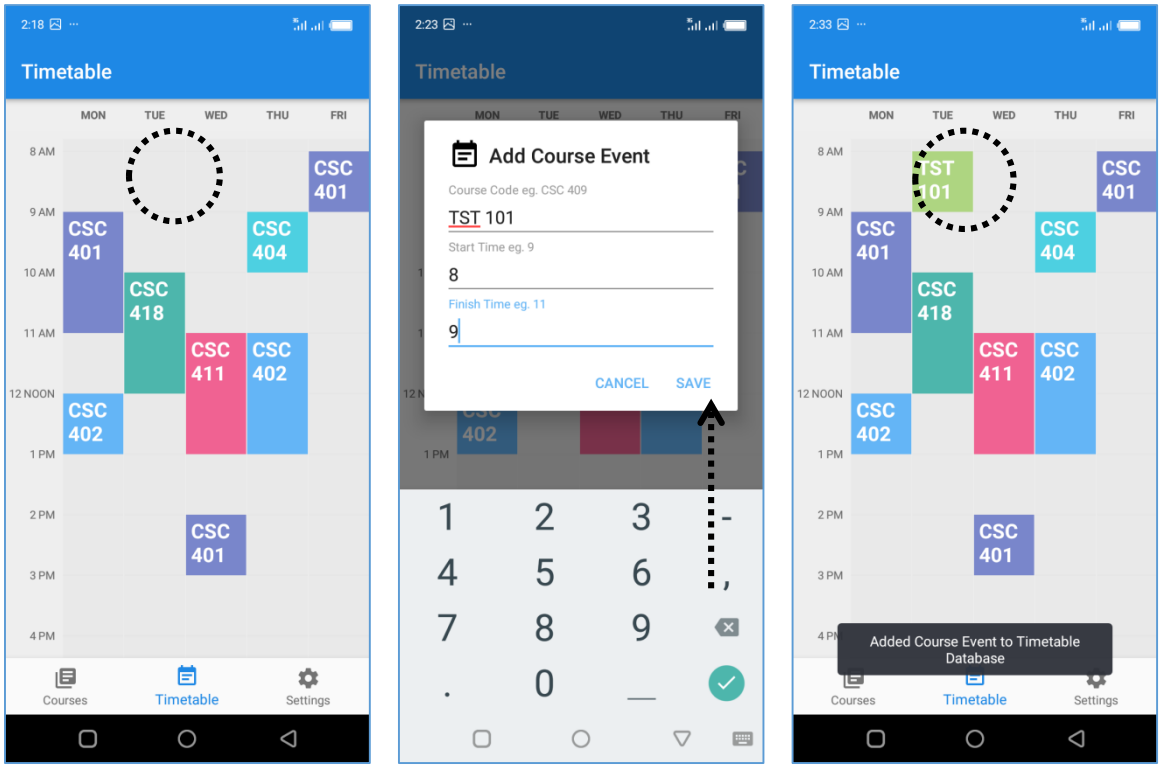


*Figure 4.18: Showing the process of **adding** a course to the timetable*

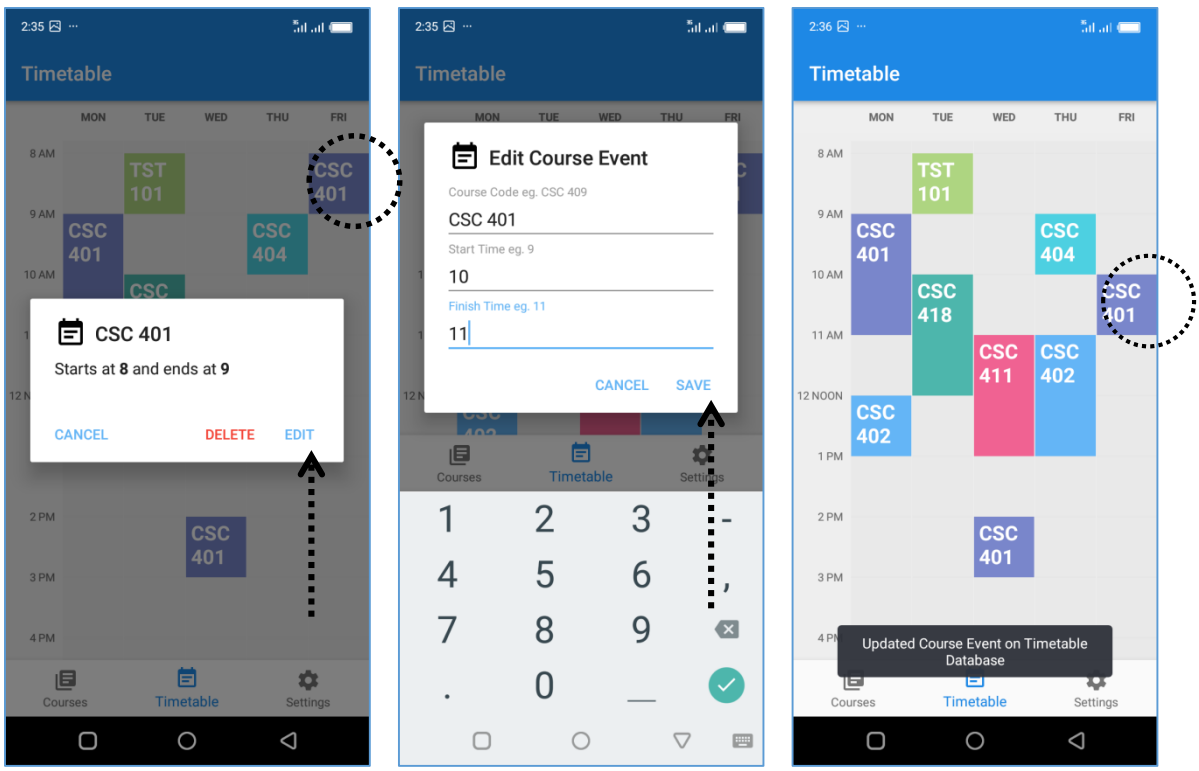


*Figure 4.19: The process of **editing** a course on the timetable*

The admin is able to add, edit and delete entries on the timetable as the need arises.

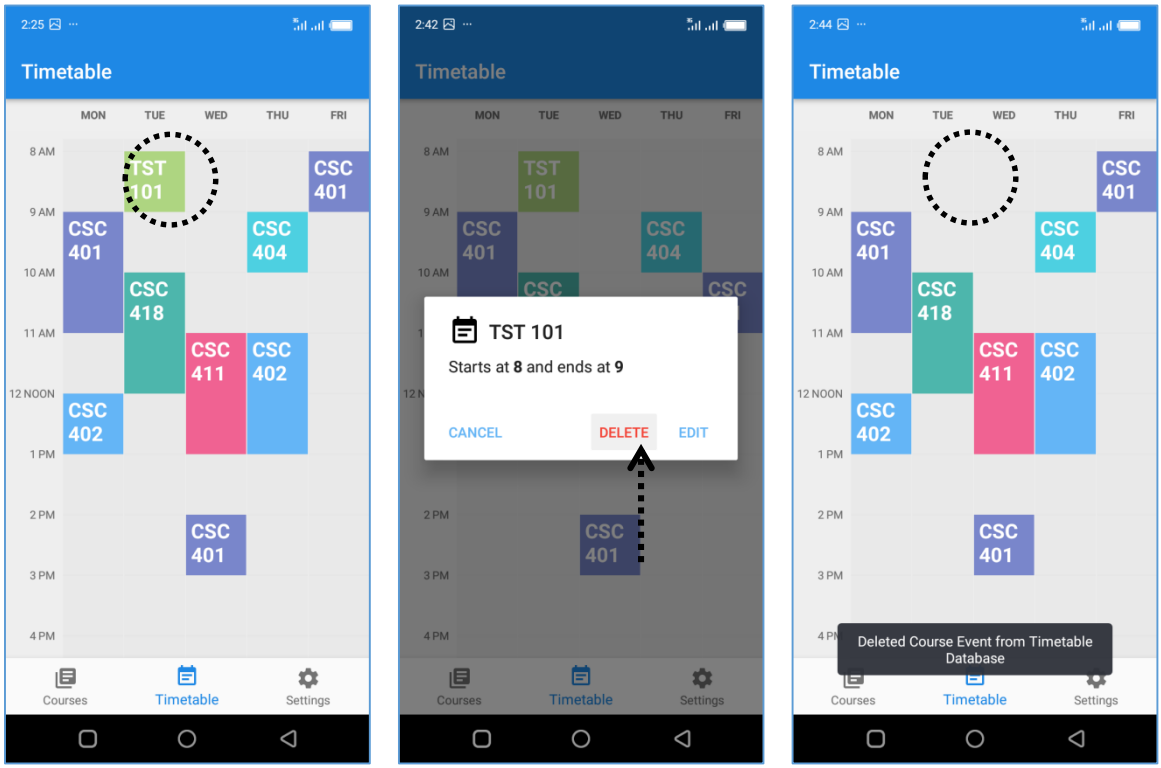


*Figure 4.20: The process of **deleting** a course from the timetable*

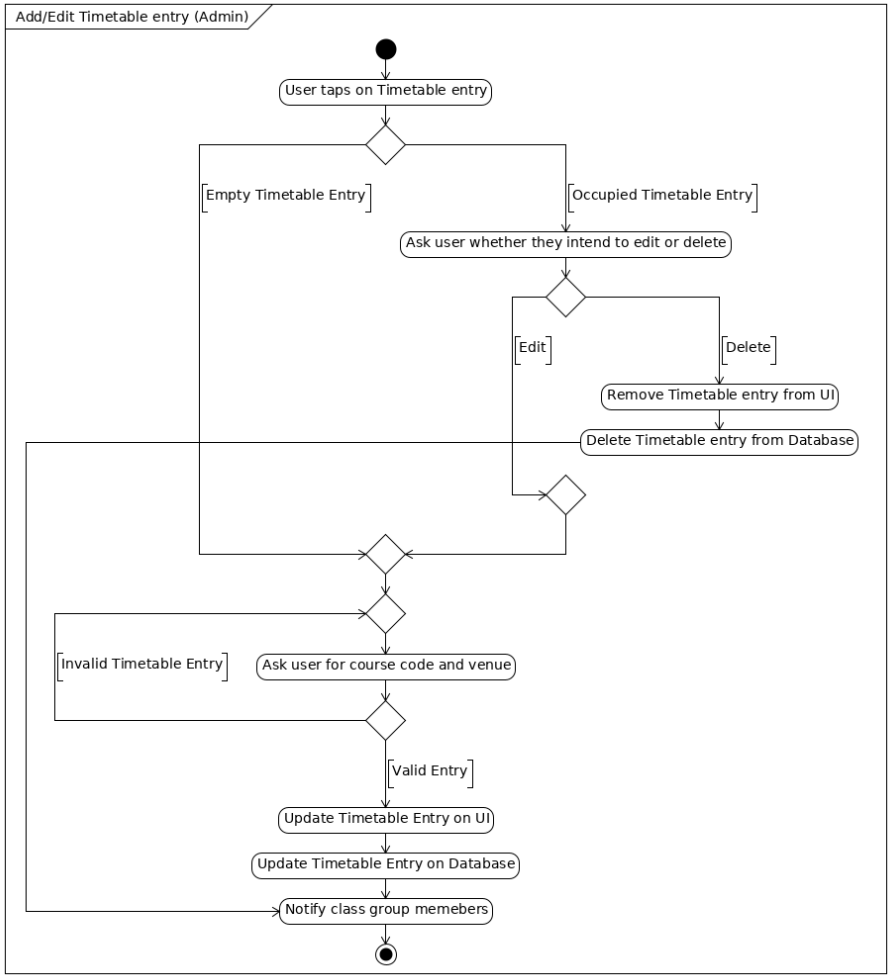


Figure 4.21: Activity Diagram for Adding or Editing a Timetable Entry

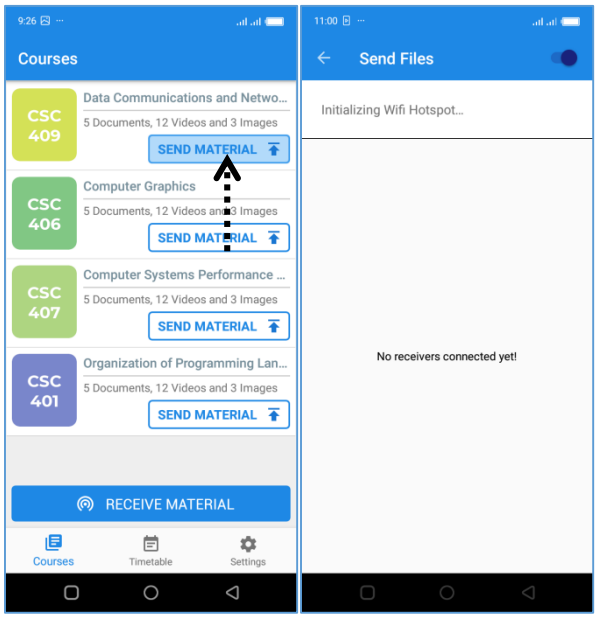
4.14 Sharing Course Material Without Data Usage via P2P Networking

One of the most important features of this application is that it allows users to send course materials to one another without a data plan. This is achieved by peer-to-peer networks. I shall enumerate the process through which the devices connect to one another and how they transmit data over the peer-to-peer network created with the help of Wi-Fi technology.

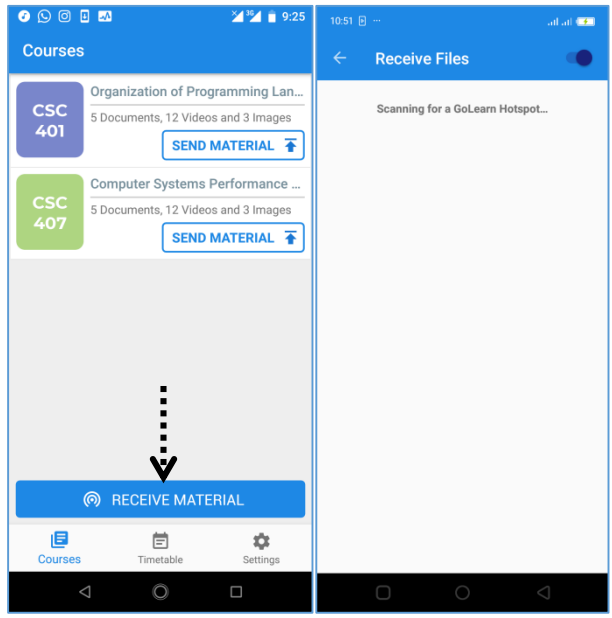
Table 4.4: Process of sending and receiving course material with the P2P technology

Sending Device	Receiving Device
When the 'Send Material' button is tapped the	We then tap 'Receive Material' on the receiving device. It then begins scanning for potential Wi-

sending device creates a Wi-Fi hotspot.

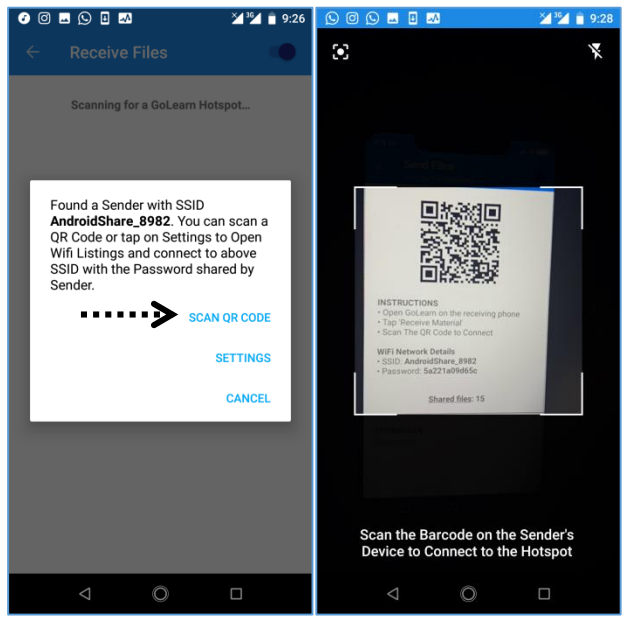


Fi hotspots

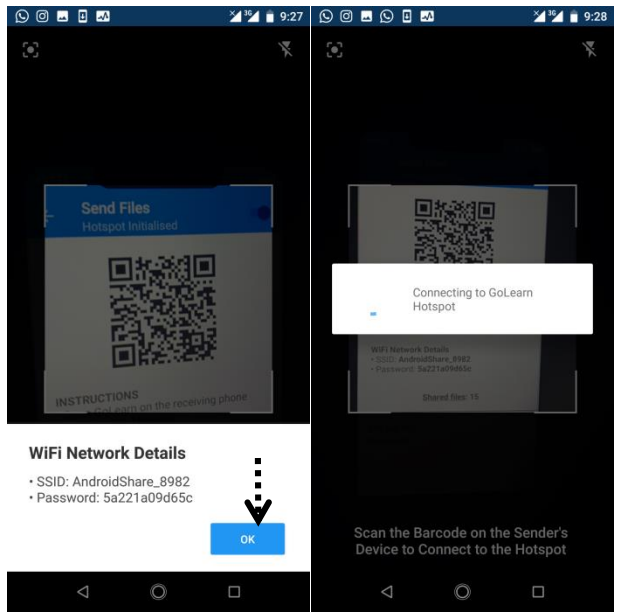
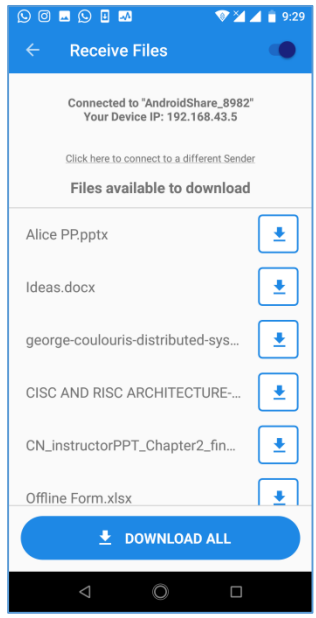


The sending device generates and displays a QR Code which encodes the SSID and Password of the Wi-Fi Hotspot.

We then tap on 'Scan QR Code' and scan the QR Code on the receiving device.



After the QR Code has been scanned the

	network details are extracted from it. 
The receiving device has now connected to the sending device and files can be shared 	The receiving device now displays all the files shared by the sending device. They can all be downloaded at once or one by one. 

4.15 Justification of Programming Languages and Technology Used

During the process of developing any system, there are development tools and technologies that one must use. Table 4.5 below outlines them.

Table 4.5: Technologies used to develop the system and the justification for using them

Tool/Technology Category	Tool/Technology Used	Justification
Programming Languages	Java and Kotlin	I used Java for the initial part of my app implementation because it was the only Android studio language that I was familiar with. I began to use Kotlin later on because it had a more straightforward syntax and better interfacing with views. It also helped me reduce NPEs (Null Pointer Exceptions) which cause about 30% of android app crashes.
Version Control Systems	Git with Github	Every time that I reached a major milestone in my implementation I committed my code changes to my Git repository.
UI Design Tools	Inkscape	I used Inkscape to design the mobile application's icon and other graphical assets. I used Inkscape because it was the graphical design tool that I am the most familiar with.
	Figma and XML	I used Figma to design how each app screen would look before converting the designs to XML for my mobile application. Figma is the UI design tool that I am

		the most familiar with.
Text Editors	Visual Studio Code	VS Code is lightweight and works well with JavaScript. I also have considerable experience using it.
Integrated Development Tools (IDEs)	Android Studio	Android Studio allows me to do almost all things android in one place. I can write code, import assets, design views, run and test my apps. It is a truly integrated solution.
UML Tools	Umlet and Microsoft Visio	I used Umlet mainly because I was on Ubuntu and it seemed the best choice. It had a decent learning curve as I was able to do fairly complicated UML diagrams after just a few hours of training. I began to use Microsoft Visio when I began typing my documentation on Windows.
UX Design Tools	-	-
Testing Frameworks	Espresso	I used Espresso because it was easy to set up and let me run almost completely exhaustive test via my UI. I was able to pin-point and fix several crashes this way.
Code Formatting and Management Tools	-	-
Debugging Tools	Android Debugger (ADB)	I used ADB because it is part of Android Studio and works well with it.

Performance Profilers	Android Studio Profiler	I used the Android Studio Profiler because it is part of Android Studio and works well with it.

Chapter Five: Testing and Evaluation

5.1 Preamble

This chapter outlines how the system was tested and evaluated in order to ensure that it functioned as expected and produced results which were accurate and precise, and met user requirements.

The purpose of testing is to formally challenge the functioning of a program, application or system - under controlled conditions - specifically to detect errors or unexpected system responses in order to manage risks of developing, producing, operating and sustaining systems.

Every software development project has the have a testing and evaluation phase in order to ensure that the software product has met user requirements and reached an adequate level of quality.

Software products can be tested in many ways. For my system, I ran both manual tests and automated tests.

What follows is a report on my testing and evaluation process, the steps I took, the types of testing and evaluation methodologies employed, the tests that I actually ran and my results.

5.2 Testing Methodology

As I said earlier, this project employed the Agile Method of software development. I have already explained what that entails, so I will not do that here.

Due to this Agile Philosophy, I had outlined the tests I would run after the desired functionality had been conceptualized. My testing ran in parallel with my implementation phase. Immediately I implemented and feature, I would run tests to ascertain that it was functioning properly and did not introduce any bugs into the older components. If I noticed that a bug was introduced as I did several times, I would do and rollback to my previous Git commit and analyze this changes. This made it easy and straightforward for me identify the offending code and rectify the issue quickly.

Below I have outlined my features and how I tested them.

Table 5.1: Table of the Functionality and the tests run for the system

Feature Set	Functionality	Description and Remarks
Courses	View Course List	**Description** This was the first piece of functionality I implemented for this feature set. It simply shows the user all the courses that are available to them.

		Remarks This feature did not have any integration problems.
	Delete Course	**Description** This functionality allows the user delete a course and all the course materials within it. **Remarks** When first implemented the folders were not being deleted completely. I had to implement a recursive method that went to the deepest folder in the course folder and recursively deleted the files in a bottom-up fashion.
	Add Course	**Description** This functionality allows user to add a course to the app. This course then reflects on the course list. **Remarks** I had no issues implementing this feature
Course Material	View Course Material List	**Description** This functionality allows the user to see all the course materials that have been added to a specific course. **Remarks** 1. The first issue that I had with this feature was that It was displaying the wrong image for file types. I rectified this by using both the file extension and mime type to determine the type of file. 2. The second issue was that the content title was not being presented in a 'nice-looking' manner. There were underscores and hyphens all along the title. I fixed this by replacing the underscores and hyphen no surrounded by

		spaces with one space.
	Delete Course Material	**Description** This functionality allows a user to delete a single course material from a course **Remarks** The issue that I had while implementing this was that I had not requested the appropriate file system permissions (WRITE_EXTERNAL_STORAGE) from the activity in which the functionality was implemented. I rectified this by simply requesting the permission and making sure that it had been granted before continuing.
	Add Course Material	**Description** This functionality allows users to add course material file (PDF, Office Document, Video, etc.) to a course. **Remarks** The issue I had with this was that course material could be added without giving it a title. I rectified this issue by setting the title to the name of the chosen file without an extension if the title space was empty and preventing submissions without a title from being submitted.
	View Course Material File	**Description** This functionality allows users to view course material by rendering the concerned file (PDF, Office Document, Video, etc.). **Remarks** The issue that I had with this feature was I had not declared the content resolver permissions properly. This was preventing the viewer from accessing the file it was to render. I resolved this by declaring my content resolver properly.

Timetable	View Timetable	**Description** This functionality shows the user their timetable. It displays all the courses being taken and their lecture times. **Remarks** 1. The first issue that I had with this was that the course times weren't being displayed properly. For example, 12 Noon, was being displayed as 0. I rectified this by writing a procedure to differentiate AM from PM and give the correct/viewer friendly time. 2. If the screen didn't have enough space, the course code did not show fully. I rectified this by modifying the font size of course code view as the screen size changed.
	Delete Timetable Entry	**Description** This functionality allows users to delete a timetable entry. **Remarks** The timetable view did not remove the deleted course immediately. I rectified this by persisting the deletion then reloading the entire fragment any time that a timetable event was deleted.
	Add/Edit Timetable Entry	**Description** This functionality allows users to add or edit a timetable entry. **Remarks** The timetable view did not add the new course immediately. I rectified this by persisting the timetable update then reloading the entire fragment any time that a timetable event was added or edited.

Settings	View Settings List	**Description** This functionality allows users to change personal and group settings/preferences. **Remarks** I did not have any issues implementing this functionality.

5.3 Testing

This section covers the testing of the entire software product. It deals with concept testing and software testing.

5.3.1 Concept Testing

At the highest level, the project had to be tested at the 'proof-of-concept' point of view. This involves asking the question of whether the basic idea works. That is, does the

concept of creating a progressive cloud based course management mobile app actually work?

I was able ascertain that my concept was sound and not based in fantasy.

The reasons for this are;

1. Many students do not access online course management systems because they do not work on their smartphones and they don't have PCs. By being mobile first my application caters to this problem.
2. Many students do not use online systems because they fear that such systems won't function properly when they do not have an internet connection. My application functions properly without and internet connection and can allay these fears.

5.3.2 Software Testing

Software testing involves both verification and validation (Somerville, 1996). Verification involves checking that the program conforms to its specification, while validation involves checking that the program as implemented meets the expectations of the user. Static checking techniques include program inspections and analysis. Dynamic techniques (tests) involve exercising the system. The testing process normally has five stages:

1. Firstly, individual units are tested in unit testing.
2. Module testing tests modules (usually a collection of dependent units).
3. Sub-system testing tests collections of modules and often exposes sub-system interface mismatches.

4. System testing tests the system as a whole and finally, there is
5. Acceptance (User) testing.

5.3.2.1 Unit Testing

A unit is the smallest testable portion of a system which can be compiled, loaded and executed. This kind of testing allows each module to be evaluated separately. The aim is to check if the components are fulfilling their individual functionalities or not.

5.3.2.2 Integration Testing

In integration testing, different software modules are combined and tested as a group. Integration testing checks the data and control flow from one module to another.

For my integration testing, I launched each module as a separate activity and checked that all the sub-components of the module functioned and interacted with each other properly. This phase was 75% automated using Espresso.

5.3.2.3 System Testing

This is performed on a complete, integrated system. It allows checking system's compliance to the specified requirements. It tests the overall interaction of components and involves load, performance, reliability and security testing. System testing was the final phase of testing that I personally carried out.

For my system testing I used to the to perform all its use-cases (as I have outlined earlier). This phase was about 70% automated using Espresso.

5.3.2.4 Acceptance Testing

Acceptance testing is carried out to find if the requirements of a specification has been met when the delivery is made

For this phase, I gave the application to students to use after which I asked them a list of questions about their experience.

I gained the following positive insight which left me satisfied with the direction in which I had taken the application.

1. 71% of respondents liked that the application was mobile based and ran on their smartphones.
2. 80% of respondents regarded the app as user friendly and simple to user.
3. 73% of respondents liked the app design.
4. 89% of the respondents regarded the peer-to-peer sharing feature as 'important' or 'very important'.
5. 66% of respondents liked the cloud syncing timetable.

6. 54% thought that it was very important that the application was able to work well while offline.

5.3.3 Platform Tests

5.3.3.1 Android Platform Tests

As the android application was one of the primal components of my system, I had to run thorough tests on multiple Android OS versions. On the android platform over 80% of smartphone owners do not run the most recent version of the OS. This makes it pertinent to test and ensure that the application runs on many of the previous OS versions.

By supporting OS versions all the way back to Android 4.1.x Jelly Bean, API Level 16, I was able to reach 99% android smartphone coverage for my app.

A problem caused by supporting all the way back to 4.1.x was that OS versions 4.1.x, 4.2.x and 4.3.x did not support the QR code scanning feature. This is because the Android camera API on those versions doesn't provide and efficient way to get and process the coming frames in real-time. For this reason, for devices running those versions, the user has to type in the password to the Wi-Fi Hotspot manually. I am currently looking for ways to circumvent this limitation.

Getting physical devices running the 13 API Levels that I was supporting would have been arduous and time wasting. I instead ran my tests using the Espresso framework on Firebase Cloud Testing. With this service, I could simply write my tests and specify the OS versions that I wanted to support and the tests could be run.

As testing is iterative, I would run the tests fix errors and bugs that popped up and run them again. This iterative process was able to reduce the number of problems by 89% (from150 to 16).

On Android OS Version and API Level, the following tests were run:

1. **Installation Tests** - This involved checking that the application was successfully installed on the device.
2. **Permissions Tests** - This involved checking that the application was properly requesting the permissions that it required for every activity in which they were required.
3. **Networking Tests** - This involved checking that the device could create Wi-Fi Hotspots. It also involved checking that the device could connect to and disconnect from Wi-Fi networks as required.
4. **View Tests** - This involved checking that the views for each activity and fragment were rendered properly. The testing service allowed me to download videos and images to ascertain that the view looked up to spec.
5. **Performance Tests** - This involved checking that the application was running in a timely manner. Tests were run to verify that the application initially loaded quickly, connected to networks quickly and responded to user inputs within reasonable time periods. Very specifically, I checked that QR Codes were scanned in less than 3 seconds to give a good user experience.

Table 5.2: Operating System versions and corresponding test results

Android Version	API Level	Test Remarks	Results
4.1.x Jelly Bean	16	All tests ran successfully but the app couldn't support QR Code scanning because the minimum supported version was 4.4 KitKat.	Successful
4.2.x Jelly Bean	17	All tests ran successfully but the app couldn't support QR Code scanning because the minimum supported version was 4.4 KitKat.	Successful
4.3.x Jelly Bean	18	All tests ran successfully but the app couldn't support QR Code scanning because the minimum supported version was 4.4 KitKat.	Successful
4.4 - 4.4.4 KitKat	19	All tests ran successfully.	Successful
5.0 Lollipop	20	All tests ran successfully.	Successful
5.1 Lollipop	21	All tests ran successfully.	Successful
5.1.1 Lollipop	22	All tests ran successfully.	Successful
6.0 Marshmallow	23	All tests ran successfully.	Successful
7.0 Nougat	24	All tests ran successfully.	Successful
7.1 Nougat	25	All tests ran successfully.	Successful
8.0.0 Oreo	26	All tests ran successfully.	Successful
8.1.0 Oreo	27	All tests ran successfully.	Successful
9.0 Pie	28	All tests ran successfully.	Successful

5.3.3.2 PC Platform Tests

The software was developed on a Linux PC running Ubuntu 16.04 (and later on Ubuntu 18.04). Due to the fact that I continuously upgraded my Android Studio IDE to get the latest

features I developed the application on Android Studio 3.1, 3.2, 3.3 and finally 3.4.

5.4 Evaluation

This section covers the evaluation phase of the project. It reviews the objectives and requirements of the project and evaluates the implementation of the system. It answers the question as to whether the system works as it should. Furthermore, it discusses the software and user evaluation of the project.

5.4.1 Requirements Evaluation

Table 5.3: Table of user requirements and their completion status

Requirement	Status
Allow Admin to Add/Edit/Delete Courses	Completed
Allow Admin to Add/Edit/Delete Course Material	Completed
Allow Admin to Add/Edit/Delete Timetable Entries	Completed
Allow Users to View Course Material	Completed
Allow Users to Share Course Material (via P2P)	Completed
Notify Users of New Course Material	Completed
Allow Users to Download Course Material	Completed

5.4.2 User Evaluation

5.4.2.1 Population Selected for Testing

The developed prototype was tested by different people through alpha testing. These people include my supervisor, five students, myself and a Geology Professor.

5.4.2.2 Usability (Beta) Testing

Usability testing allows one to see how easily a software product can be used by testing it with ‘real’ users. To determine the user experience, users were asked to complete certain tasks. The users’ actions were observed to see if and where they would encounter problems while using the application. Whilst traditional testing might be undertaken by a developer, designer or project manager, usability testing helps remove “creator’s bias” by collecting feedback directly from the end-user.

The intention of conducting these tests is to observe if users were able to complete tasks (guided by the application’s use cases) adequately while documenting errors or problems they may have encountered. The experiment recorded the problems encountered the across the testing group. The aim of this was to identify changes that could be made to the application to ensure a more positive user experience, and thus, overcome the usability issues encountered.

During the beta testing exercise, 30 students were given the mobile application prototype to use. Table 5.4 below shows the tasks each user was expected to accomplish, the number of participants carrying out each task and the number of users that successfully completed each task.

Table 5.4: Users' Task description for mobile application

Tasks	Number of Participants that carried out the task	Number of Participants that successfully completed the task
Register an account	30	29
Sign in to account	30	30
Add a course	30	30
Delete a course	30	30
Add course material	30	29
Delete course material	30	30
Add a timetable event/entry	30	30
Edit a timetable event/entry	30	30
Delete a timetable event/entry	30	30
Edit User Information	30	30
Edit University Information	30	30

Table 5.5 below provides a summary of the findings in the beta test of the mobile and web application respectively. It shows the user id of the tester, the various complain from the users and the researcher's observation during the test.

Table 5.5: Complaints brought up by the beta testers of the mobile application

User Identifiers	User Complaint	Observations and Remarks
#5, #8	These users said that user interface was too colourful	The colours used as the background for the course codes in the main course list are progressively generated based on the name of the course. I fixed this by reducing the number of

		possible colours from 256 to 16.
#9	This user requested that the application supported more languages.	This feature addition will have to fall under possible future work but it does not seem very important as 100% of the students can speak English.
#15, 22	These users wanted the application to display PDF Documents in a mobile friendly format since the text in some documents looked too small.	This feature request seems important but the problem lies in the technical implementation. PDF documents that were not created with mobile device reflow in mind can get scattered if they are forcefully reflowed rendering the whole document incorrect. Implementing a software that can consistently and effectively reflow **all** kinds of PDF documents can be a separate project in its own right given the difficulty in such an endeavour.
#7	The users wanted a provision (inside the application) to provide their feedback.	I added a feedback screen to the beta version of the application.

5.4.3 Performance Evaluation

In order to ascertain that the application performed to the required standard, performance evaluation had to be carried out. I did not want a case where the application offered the needed functionality but the user experience was ruined because the application was not fast enough for its users.

Good performance is even more pertinent in this day and age as users have been ‘spoilt’ with applications that load in under half a second and give results almost instantly.

I will outline the performance evaluation metrics employed in evaluating the system. Note that all performance evaluation tests were carried out in an automated environment

5.4.3.1 Response Time

The response time experienced by users while navigating the application is a very important component of the user experience (UX). When it takes a long time for the user’s input to be acknowledged user’s feel the app is slow. I also employed animations while switching from activity to activity in order to make the transition from various screens seem more fluid and quick.

Table 5.6 below shows, the startup time and response time when transitioning between various screens. All the tests were run 10 times and averaged to minimize the effect of outlier values.

Table 5.6: Startup time and Response time (measured in seconds) across application screen

transitions

Device	Samsung		Infinix		Tecno	Gionee		HTC	Huawei	Google Pixel
Model (Android OS Version)	*S8 (7)*	*S9 (8)*	*Hot 6 (7)*	*S3X (8)*	*Camon 11 (7)*	*S11 (7)*	*M7 (8)*	*Desire 12s (8)*	*Y9 (8)*	*3 XL (9)*
Startup Time	0.76	0.65	1.45	1.34	1.85	1.58	1.52	0.9	0.79	0.69
Display Timetable Screen	0.35	0.39	0.57	0.51	0.67	0.88	0.78	0.5	0.45	0.33
Display Courses Screen	0.24	0.25	0.44	0.46	0.56	0.73	0.66	0.42	0.37	0.21
Display Settings Screen	0.15	0.19	0.37	0.31	0.47	0.68	0.58	0.3	0.25	0.13
Display Edit University Info Screen	0.39	0.42	0.55	0.56	0.69	0.83	0.71	0.52	0.49	0.37
Display Edit User Info Screen	0.32	0.36	0.54	0.59	0.62	0.88	0.78	0.53	0.43	0.31
Display Add Course Screen	0.39	0.42	0.55	0.56	0.69	0.83	0.71	0.52	0.49	0.35
Display Add Course Material Screen	0.35	0.39	0.57	0.51	0.67	0.88	0.78	0.5	0.45	0.33
Display Sign in/Register Screen	0.24	0.25	0.44	0.46	0.56	0.73	0.66	0.42	0.37	0.21
Display Welcome Screen (when starting up)	0.15	0.19	0.37	0.31	0.47	0.68	0.58	0.3	0.25	0.13
Display Welcome Screen (after	0.24	0.25	0.44	0.46	0.56	0.73	0.66	0.42	0.37	0.21

signing out)										

5.4.3.2 Upload Speed

One of the most important performance metrics for the system is the speed of uploads. Before any learning material is uploaded it is compressed. A close look at Table 5.4 below shows the upload times for Word Documents were generally lower than those of PDFs at the same size. The reason for this is that the Word Documents had much better compression ratios than the PDFs; averaging 30% vs 2%.

For each the result in each table cell the test was run 10 times. The average value of the 10 tests is what was used to fill the table.

Table 5.7: Upload performance for various file types and sizes

		2G/EDGE	**3G/HSPA**	**4G/LTE**	**Poor Wi-Fi**	**Average Wi-Fi**	**Best Wi-Fi**
		50-500 Kbps	500Kbps-10Mbps	10Mbps-100Mbps	100Mbps-500Mbps	500Kbps-10Mbps	10Mbps-100Mbps
File Type	**File Size**						
PDF	1MB	3.75s	0.6s	0.2s	3.3s	0.55s	0.1s
PDF	5MB	20.4s	2.8s	1.2s	18.7s	2.4s	0.5s
PDF	15MB	58.7s	8.0s	2.0s	54.3s	7.7s	1.3s
Word	1MB	2.9s	0.4s	0.15s	2.3s	0.3s	0.1s
Word	5MB	13.4s	2.0s	0.8s	12.8s	1.8s	0.7s
Word	15MB	38.5s	6.7s	1.7s	36.2s	5.7s	1.1s

5.5 Discussion

The testing of the prototype covered installation, functionality and usability testing. The installation testing was carried out to verify that installation of the mobile application on android devices was successful. The functionality testing was carried out to verify that each function, class and section of the software application operated in conformance with the requirement specifications.

The testing was majorly black-box testing as it was not concerned with the source code of the prototype. Each and every functionality of the system was tested by providing appropriate and inappropriate inputs, verifying the output and comparing the actual results to the expected ones.

The usability testing was performed by the users (students) with the goal of identifying any usability, UX, or UI problems.

The complaints from the users of the mobile application included the lack of a screen for providing feedback, the app being too colourful and the application not supporting native languages. The first two complaints were implemented while the last one was mostly ignored.

Findings from the first phase of the beta testing showed that the users were able to use the application with ease. This is a very important characteristic of a mobile learning application.

Other minor complaints and observations were noted and the required changes were implemented.

Chapter Six: Summary, Conclusion and Recommendation

6.1 Preamble

This chapter attempts to offer up a worthwhile conclusion to this research project. I shall also evaluate how well the project was executed based on the objectives set forth in Chapter One. I shall also give a brief summary of my work and offer a helpful background for future work on this and similar research topics.

This chapter brings this research work to an end. The chapter will summarize all the previous chapters and restate the problem the work aimed to tackle along with the objective and solution to all sub-objectives.

In Chapter One, the problem statement stated as follows: “There is no cloud and mobile based learning material management system available to the students of the University of Jos”. Subsequent chapters tried to address this problem by reviewing academic and other literature, gathering requirements, designing and developing a prototype of the system, and finally testing and evaluating the implemented solution.

6.2 Summary of Findings

This subsection will summarize the findings from the whole research project by highlighting the primary points. This summary assumes that the reader has an at least rudimentary understanding of the preceding chapters.

In the first chapter, cloud learning systems and their composition was discussed. The problem was stated. The aim of the research was identified and the primary and secondary objectives were outlined. From this perspective, Chapter Two began by going through academic and other literature and reviewing of the ideas and concepts behind cloud technologies and education. What followed was a discussion on the influence of IT on education taking into consideration the technologies involved. Thereafter, different kinds of cloud-based learning applications were reviewed, and their primary features and use contexts were identified.

Chapter Three started by describing the case study for this research, using the findings from the previous chapters. Through surveys, I aimed to glean the current requirements of a cloud-based learning system. Findings from the surveys conducted showed that the vast majority of respondents used mobile devices very regularly. This fact indicated that it was of great importance for the system to be both mobile-based and cloud-based.

Chapter Four started by presenting an analysis of the proposed system before going on to design and finally implement the system. Chapter Five tested and evaluated the developed system to determine if the specifications outlined in Chapter Four were met. The next section shall show the accomplishment of all the objectives we discussed in Chapter One.

6.3 Accomplishment of Objectives

This section will briefly discuss how the sub-objectives were accomplished and how they collectively worked towards the primary objective. This section will further mention how the primary objectives comprehensively addressed the problem statement from Chapter One.

6.3.1 Objectives

As mentioned in Chapter One, the primary objective of this body of work is "to design and implement a system that will allow students learn more easily and efficiently using mobile and cloud technologies."

Due to the generalized nature of this objective and the limited time allotted for work on this project, these secondary objectives became necessary:

- To identify the most critical features of such a system as well as existing systems that offer similar functionality.
- To gather all the requirements of such a system in order to make decent progress in the work and form a basis for future research and implementation.
- To develop a system that facilitates students' easy and effective access to learning material.

6.3.1.1 First Sub-objective

The first sub-objective is "to identify the most critical features of such a system as well as existing systems that offer similar functionality." The method employed to meet this sub-objective was an extensive review of relevant literature of existing systems. Through this, it was discovered that amongst other features, a cloud-based learning system should allow its users to access their learning material from anywhere and at any time.

6.3.1.2 Second Sub-objective

The second sub-objective is "to gather all the requirements of such a system in order to make decent progress in the work and form a basis for future research and implementation." This was realized by means of online surveys (the qualitative route). This gave more insight into what users desired from such solutions and opinions regarding the functionality which the system should provide to maximize positive learning outcomes. These findings served as a basis for the design and implementation of the system.

6.3.1.3 Third Sub-objective

The third sub-objective is "to develop a system that facilitates students' easy and effective access to learning material." In Chapter Four, the system's components were shown and descriptions were provided for each one. Also outlined, were how these components related to one another to form the system as a whole. The system was then implemented based on the requirement and other details that formed the proposed solution.

Adequate testing was carried out to ascertain that this sub-objective was carried out well.

6.3.2 The Objectives with Regards to The Entire Project

Therefore, it is important to note that the three sub-objectives collectively addressed the primary objective of this research work, which is “to design and implement a system that will allow students learn more easily and efficiently using mobile and cloud technologies”. The primary objective, in turn, comprehensively addressed the problem statement which is: “There is no cloud and mobile based learning material management system available to the students of the University of Jos”.

6.4 Suggestions for Further Research

This research work is currently limited to the Android Operating Systems. Although, I demonstrated that process of getting the P2P sharing feature to work on iOS in a user-friendly manner is challenging, I would recommend periodical studies be carried out on the feasibility of that endeavor.

6.5 Conclusion

This project has identified the critical aspects needed to develop a mobile cloud-based learning system. The features of this system were determined and finally an initial implementation that served as a proof of concept was designed and showcased. Therefore, the research can be said to have achieved its target objective. It is hoped that the use of this system will be stretch beyond the University’s environment.

References

1. Alizadeh, M., & Hassan, W. H. (2013). Challenges and opportunities of mobile cloud computing. In 2013 9th International Wireless Communications and Mobile Computing Conference (IWCMC) (pp. 660–666). http://doi.org/10.1109/IWCMC.2013.6583636
2. Bathon, J. (2013). The fine print on cloud computing. T H E Journal (Technological Horizons In Education), 40(9), 23–26.
3. Bedell, C. (2014). The state of cloud computing in K-12. K-12 TechDecisions, Nov 2014. Retrieved from http://www.k-12techdecisions.com/images/pdfs/TD_K12_DeepDive_Cloud_Nov14.pdf
4. Bundsgaard, J. (2010). Faglighed og digitale læremidler i undervisningen - Forskning - Aarhus Universitet. Dansk pædagogisk tidsskrift, 4(10), 15–23.
5. Carroll, M., Merwe, A. van der, & Kotzé, P. (2011). Secure cloud computing: Benefits, risks and controls. In 2011 Information Security for South Africa (pp. 1–9). IEEE. http://doi.org/10.1109/ISSA.2011.6027519
6. Chandra, D. G., & Borah, M. D. (2012). Cost benefit analysis of cloud computing in education. In 2012 International Conference on Computing, Communication and Applications (pp. 1–6). http://doi.org/10.1109/ICCCA.2012.6179142
7. DiNucci, D. (1999, April). Fragmented Future. Print, 32, 221–222.
8. Egelund, N. (2014). PISA problemløsning : danske unge i en international sammenligning. Dafolo. Retrieved from https://bibliotek.dk/da/work/870970-basis:51171810
9. EMU Denmark's learning portal. (2015). It og medier - vejledning. Retrieved September 23, 2016, from http://www.emu.dk/modul/vejledning-det-tv%C3%A6rg%C3%A5ende-emne-it-og-medier
10. Fibiger, J. (2008). Litteraturens tilgange. Academica.
11. Hung, C., & Lin, C. (2013). An integrated cloud application in school knowledge management implementation. In Proceedings of International Conference on Law, Humanities and Education. Penang, Malaysia. Retrieved from http://psrcentre.org/images/extraimages/2%20813555.pdf
12. Levinsen, K. T. (2011). Udfordringer i netværkssamfundet - Forskning - Aarhus Universitet. Dansk Universitetspaedagogisk Tidsskrift, 11, 20–25.
13. Liberati, A., Altman, D. G., Tetzlaff, J., Mulrow, C., Gotzsche, P. C., Ioannidis, J. P. A., … Moher, D. (2009). The PRISMA statement for reporting systematic reviews and meta-analyses of studies that evaluate healthcare interventions: explanation and elaboration. BMJ, 339(jul21 1), b2700–b2700. http://doi.org/10.1136/bmj.b2700
14. Madsen, M. S. (2014, June 17). Google Docs smidt ud af svenske skoler. Computerworld. Retrieved from http://www.computerworld.dk/art/231214/google-docs-smidt-ud-af-svenske-skoler
15. Mahalingam, T., & Rajan, A. V. (2013). Cloud and mobile computing: Affordances of the 21st century teaching and learning. In 2013 International Conference on Current Trends in Information Technology (CTIT) (pp. 125–128). IEEE. http://doi.org/10.1109/CTIT.2013.6749490
16. Masud, M. A. H., Yong, J., & Huang, X. (2012). Cloud Computing for Higher Education: A roadmap. In 2012 IEEE 16th International Conference on Computer Supported Cooperative Work in Design (CSCWD) (pp. 552–557). IEEE. http://doi.org/10.1109/CSCWD.2012.6221872
17. Moher, D., Liberati, A., Tetzlaff, J., & Altman, D. G. (2009). Preferred Reporting Items for Systematic Reviews and Meta-Analyses: The PRISMA Statement. Annals of Internal

Medicine, 151(4), 264–269. http://doi.org/10.7326/0003-4819-151-4-200908180-00135
18. P21. (2015). Partnership for 21st century learning: P21 framework definitions. Washington, DC: P21. Retrieved from http://www.p21.org/storage/documents/docs/P21_Framework_Definitions_New_Logo_2015.pdf
19. Reidenberg, J., Russell, N. C., Kovnot, J., Norton, T. B., & Cloutier, R. (2013). Privacy and cloud computing in public schools. New York: Center on Law and Information Policy. Retrieved from http://www.studentprivacymatters.org/wp-content/uploads/2014/04/Privacy-and-Cloud-Computing-in-Public-Schools.pdf
20. Spring, J. (2008). Research on Globalization and Education. Review of Educational Research, 78(2), 330–363. http://doi.org/10.3102/0034654308317846
21. Stein, S., Ware, J., Laboy, J., & Schaffer, H. E. (2013). Improving K-12 pedagogy via a cloud designed for education. International Journal of Information Management, 33(1), 235–241. http://doi.org/10.1016/j.ijinfomgt.2012.07.009
22. Sule, M. J., Li, M., Taylor, G. A., & Furber, S. (2015, September). Deploying trusted cloud computing for data intensive power system applications. In Power Engineering Conference (UPEC), 2015 50th International Universities (pp. 1-5). IEEE.
23. Thomas, P. Y. (2011). Cloud computing: A potential paradigm for practising the scholarship of teaching and learning. The Electronic Library, 29(2), 214–224. http://doi.org/10.1108/02640471111125177
24. Ullman, E. (2015). The case for cloud computing in k12. Tech & Learning. Retrieved from http://www.techlearning.com/white-papers/0009/the-case-for-cloud-computing-in-k12/69386
25. Weil, M. (2011). On Cloud Nine. THE Journal, 38(6), 46–51.