

## A Case Study on Academic Services Application Using Agile Methodology for Mobile Cloud Computing

Mahmoud M. Abu Ghosh<sup>1</sup>, Wael F. Al Sarraj<sup>1</sup>

<sup>1</sup> Faculty of Information Technology Islamic University of Gaza (IUG) Gaza, Palestine

**Abstract:** Recently, Mobile Cloud Computing reveals many modern development areas in the Information Technology industry. Several software engineering frameworks and methodologies have been developed to provide solutions for deploying cloud computing resources on mobile application development. Agile methodology is one of the most commonly used methodologies in the field. This paper presents the MCCAS a Web and Mobile application that provide feature for the Palestinian higher education/academic institutions. An Agile methodology was used in the development of the MCCAS but in parallel with emphasis on Cloud computing resources deployment. Also many related issues is discussed such as how software engineering modern methodologies (advances) influenced the development process.

**Keywords:** Cloud Computing, Mobile Software Engineering, Academic services, Agile.

### I. INTRODUCTION

Cloud computing is on-demand computing that does not reside at the user's premise. Instead, the computing resources (e.g., networks, servers, storage, services) are owned and managed by a service provider and the users access the resources via the internet [1]. Cloud computing is web-based processing, shared resources and information provide on demand to portable devices and computers to the users for processing [2]. Cloud computing providers offer their services according to several fundamental models: (a) Software as a Service (SaaS): The capability provided to the consumer to use the provider's applications running on a cloud infrastructure. The applications are accessible from various client devices through a thin client interface such as a web browser (e.g., web-based email). The consumer does not manage or control the underlying cloud infrastructure with the possible exception of limited user-specific application configuration settings, (b) Platform as a Service (PaaS): The capability provided to the consumer to deploy onto the cloud infrastructure consumer created or acquired applications created using programming languages and tools supported by the provider. The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, or storage, but has control over the deployed applications and possibly application hosting environment configurations, (c) Infrastructure as a Service (IaaS): The capability provided to the consumer for provision processing, storage, networks, and other fundamental computing resources where the consumer is able to deploy and run arbitrary software, which can include operating systems and applications. The consumer does not manage or control the underlying cloud infrastructure but has control over operating systems, storage, deployed applications, and possibly limited control of select networking components (e.g. host firewalls) [3]. Mobile cloud computing at its simplest, refers to an infrastructure where both the data storage and data processing happen outside of the mobile device. Mobile cloud applications move the computing power and data storage away from mobile phones and into the cloud, bringing applications and MC to not just smartphone users but a much broader range of mobile subscribers [4].

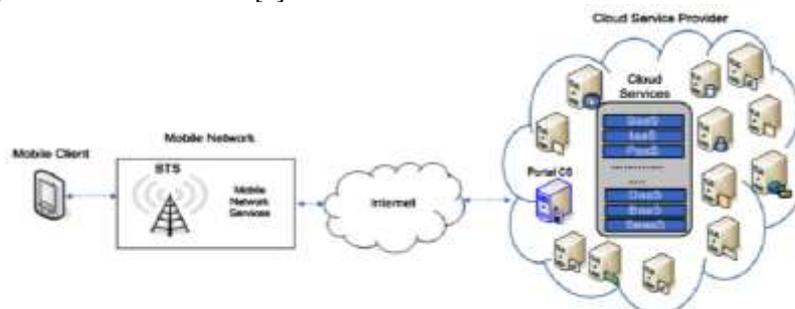


Fig. 1. Mobile Cloud Computing Architecture [5]

Mobile Cloud Computing reveals many modern development areas in the Information Technology industry. Several software engineering frameworks and methodologies have been developed to provide solutions for deploying cloud computing resources on mobile application development. Agile software development is based on a set of principles. It comprises a group of several iterative and incremental software development methods with focus on collaboration between cross-functional and self-organizing teams. Unlike traditional software development methods, agile methods focus on collaboration and interactions rather than processes [6] as shown in Fig.2 [11]. This approach enables agile software development to be able to cater to today's fast-growing industry needs by having short development lifecycle, speedy development process and constant interaction with customers [7].

Software engineering advances make revolution in mobile application, so agile methodology work is gaining large attention nowadays in mobile cloud computing [8].

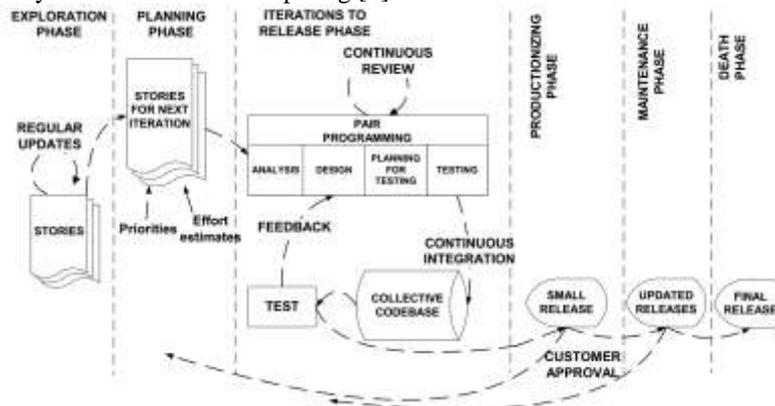


Fig. 2. Life cycle of the XP process [11]

## II. RELATED WORK

Almudarra et al. in [14] Sha Mo Cloud Service, a mobile - cloud based environment for content / media management. Design and implementation of a hybrid cloud environment was detailed with focus on integrating three tiers of the framework while preserving data security and privacy. Various aspects of development using extreme programming as an agile development methodology were discussed. The work was compared to similar projects highlighting the gains and challenges in development. The experience shows that Mobile Cloud application development can be integrated with agile development methodologies reducing the cost, time and improving software quality.

Kalathoti et al. in [15] the writer's development an CMPlayer prototype, they tried to provide a case study as a context of mixing both the technology paradigms, they built a music streaming application prototype system based on Android platform and Google App Engine to explore the impact and effect of cloud computing in mobile software environment .

AbuNaser et al. in [9] Implementing Mobile Cloud Computing Academic Services at Palestinian Higher Education Institutions (MCCAS), which is a mobile application to facilitate access. MCCAS serve the students and the academic staff of Palestinian higher education institutions. According to a study we have done, there is a strong evidence on the readiness of academic staff and students to accept and use Academic Services on Mobile in their education environment. A survey was carried out on a group of academic staff and students who used MCCAS application and answered the survey questions. The survey answered by 220 students and 51 academic staff with different levels, gender and different types of phones. The survey emphasized that MCCAS satisfies academic staff and student needs and run with high performance and compatible with all OS devices and achieved the main objective of our system.

In [9] Android Application for Islamic University Gaza (Student Portal) displaying the courses' schedule and exams' schedule for students from anywhere and anytime, also notifying the students to student lectures' schedule and exams automatically, viewing the academic information and grades report (marks transcript) for the students, providing silence schedule because most of students forget their mobile phones in normal mode during the lectures, but by the intended application, mobile phones will be automatically switched to silent mode during the lectures. But this system depends on the university Web site directly and doesn't have main activities such as semester registration and advertisement.

As illustrated in the previous related work most of them are focusing on Software Engineering methodology in particular agile model for the mobile application development: in this context we aimed to exploit their experience to guide us accomplish our work in the development of our app.

### III. ARCHITECTURE, TECHNOLOGIES, AND PROCESS

In this section, we will present the Mobile Cloud Computing Academic Services at Palestinian Higher Education Institutions (MCCAS) is a mobile application to facilities access to the Academic services to discover the impact and effect of cloud computing in mobile software environment. MCCAS serve the students, and the lecturers on Palestinian higher education institutions [9, 10] with Google App Engine in the backend. The prototype system is served as a major study subject facilitating in exploring the mobile software engineering advances in cloud computing environment.

#### A. Architecture

The MCCAS adopts the 3-tier client-server architecture and design pattern in which the presentation tier, business logic tier, data storage/access tier are developed and maintained as independent modules so that each module can be updated or replaced independently. The prototype enables application smart phone users to access to the Google cloud space for Academic services shown in Fig. 3 [15, 16,18].

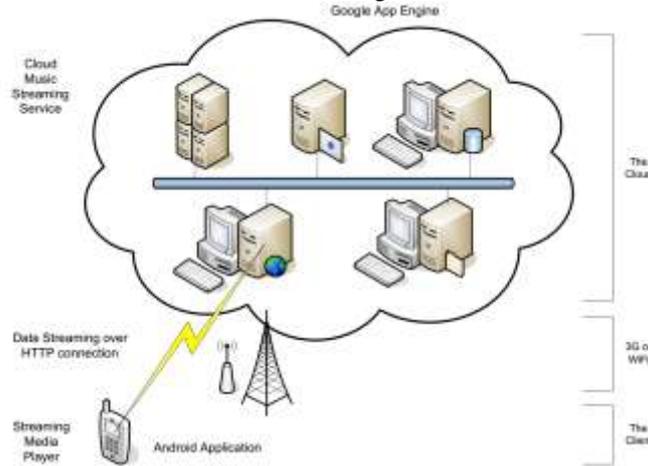


Fig. 3: Overall System Architecture [16]

#### 1. Overview

The presentation tier is a standard mobile application. This is the highest level of the application which student can use the main function as register courses, view transcript. The logic tier is across Internet between two end points: application smart phone and web application. It executes the user commands by sending a HTTP request to the cloud server end, invoking server end components execution, and retrieving back the result in a HTTP response. The data tier adopts the server as the end framework.

#### 2. System Building Block

To prototype using Academic services from the cloud, MCCAS has the following building blocks shown in Fig. 4 [16].

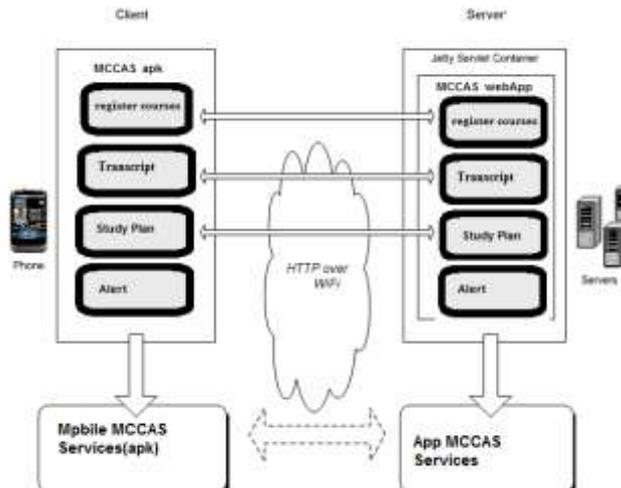


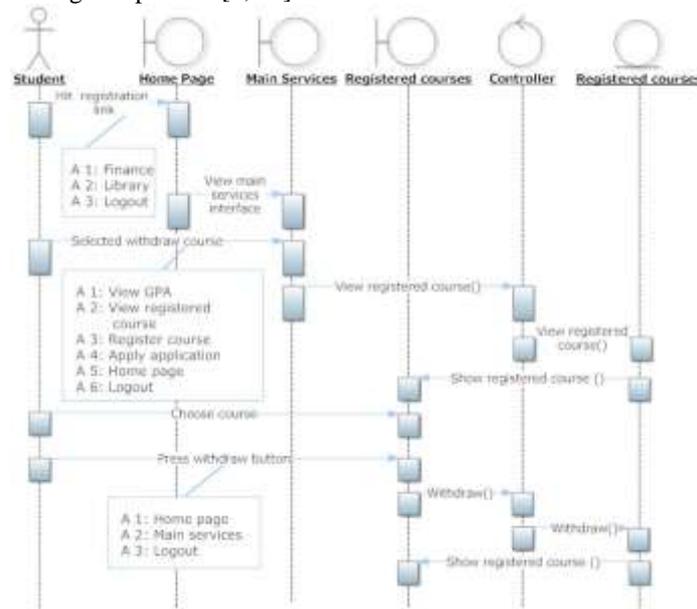
Fig. 4: System Building [16]

On the cloud space or the final servers, the MCCAS web app consists of several J2EE servlets for register courses, display the academic information for the students, also display grades report (Transcript), also the application must send an alert for the user when a new grade added, before the exam date in 3 days, and before the lecture start in 15 minutes. The application must display the Study Plan. The application must display the courses schedule, it must display the exams schedule and user authentication services.

The web app is hosted in the HTTP server. HTTP is adopted by the framework to serve as a java-based HTTP server and javax.servlet container. App MCCAS provides essential quota-based cloud computing components in which Data store services are used by MCCAS web application. On the client side, the MCCAS client app contains the corresponding activities to server end servlets, and presents a user interface to navigate the end user accessing to the services. The HTTP protocol is used by both command and data channels. By contrast, the command channel is explicitly established between the MCCAS client and server components and maintained by the MCCAS application client for command request and information exchange.

**3. System Interface**

The main functioning on the application as registration for a course flow can be shown in Fig. 5 [9]. Student hits registration link at Home Page interface. After logon the user can navigate to main services interface and select Academic services, there are alternatives services can have done as view transcript, see financial record, View Study Plan, view registered courses, register course, view the academic information, view the courses schedule, view the exam schedule, also can order academic queries, send complaints and suggestions, and finally can view laws and regulations academy. Then the user choose course from registered courses which he wants to register and press register button. register function sent to controller then to registered courses, then registered courses will be shown after register process [9, 10].



*Fig. 5: The sequence diagram for registration process [9]*

**B. Technologies**

**1. Client Side**

The client development is carried using the iOS and Android J2SE technologies and Application Framework. The MCCAS client application consists of self-defined activities for the user interface design and utilizes the system MCCAS service for Academic services. The MCCAS middle tier networking components are built upon the Apache HTTP Client library which is essentially based on the WebKit native runtime library. The server response data is wrapped in form of XML format. The rich set of the Android application framework makes it possible to build a rich featured cloud based application in mobile phone device.

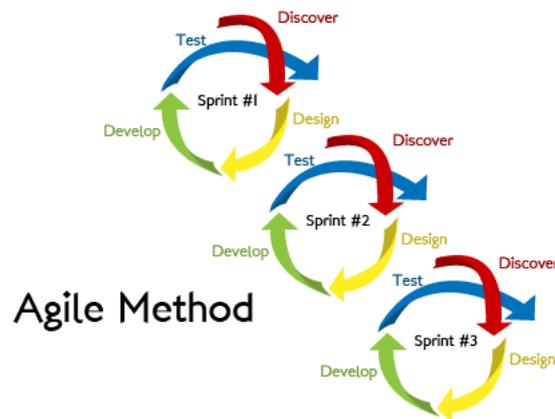
**2. Server Side**

The server development is carried using J2EE technologies (JSP/Servlet) and integrated with MCCAS services. Similar to Android platform, Eclipse IDE is the preferred development environment for MCCAS based web application.

**C. Cloud Based Mobile Software Development Process (Agile methodology)**

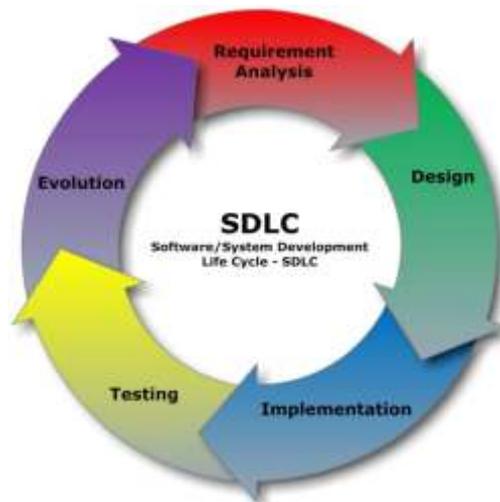
Agile methodology as Fig. 6 [8] was used to develop MACCAS application. It was promoting collaboration between both the customers and developers to retrieve immediate feedback and to acknowledge their changing requirements swiftly. It is highly iterative; developers could easily refine each of the development phases either the ones which are completed or the one which just started anytime as appropriate [9].

Agile software development methods have two main units of delivery: releases and iterations. A release consists of several iterations, each of which is like a micro-project of its own. Features, defects, enhancement requests and other work items are organized, estimated and prioritized, then assigned to a release. Within a release, these work items are then assigned by priority to iterations [11].



*Fig. 6: Cloud Based Mobile using Agile phases [8]*

Agile methodology incorporates the SDLC phases as shown in Fig. 7 [8] starting from the Planning phase up to the Maintenance phase.



*Fig. 7: Phase of SDLC in each iteration [8]*

A relationship between software development life cycle (SDLC) and some agile software development methodologies are given on Fig. 8 [11]. It should be noticed that various agile software development methodologies have different focuses in terms of a) software development project management, b) software development process, and c) software programming practices/activities/products; moreover, they are applicable to different stages of software development life cycle [11-13].

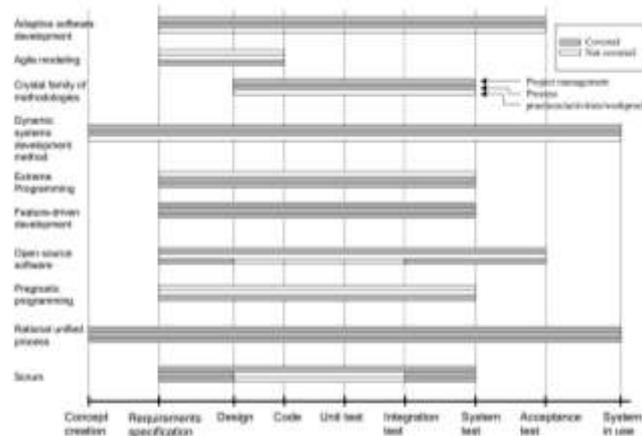


Fig. 8 : Relationship between software development life cycle and agile development methodologies [11]

#### IV. MCCAS

In this section, is using agile methodology for develop MCCAS application described, also explain each phases of the methodology and the activities which are planned to be executed and explained.

##### A. Requirement Analysis

It is the first phases, in this phase business requirements are gathered. it focuses on the project managers and stake holders to determine the requirements, the Student Academic Service Requirement. There are two main categories of the system functions we from the gathered requirements. The first category handles the user interface and the display of the functions including display the academic information for the students, courses, register the courses, grades report (Transcript), courses schedule, exams schedule, Study Plan, financial report and Order academic queries. The second category handle the notification including remind the students for lectures and exams time automatically.

##### B. System Design

In this section, we will present a high level architecture of the MCCAS (Mobile Cloud Computing: Academic Services for Palestinian Higher Education Institutions), a mobile smart phone application prototype system with Google cloud computing in the backend. The prototype system is served as a major study subject facilitating in exploring the mobile software engineering advances in cloud computing environment.

##### 1. Client Design

All the Services of MCCAS display is this interface as shown in Fig. 9 [9] academic services, library services, eLearning services, email services, exam bank services, the news and announcement, also the setting.



Fig. 9 : The main services interface [9]

Academic Information Interface as shown in Fig. 10 [9], It is display the Academic Information, as profile picture, name, Student id, University name, College name, specialized and the academic level, GPA, the hours that studied, the hours will be studied, and the registered hours



*Fig. 10 : Display the Academic Information interface [9]*

Registration services interface Through this interface can user view scheduling course, exam scheduling, financial record, semester registration, the course can be registered, Transcript, view study plan, view loan and scholarship, electronic application icon, view the honorary board for student college as shown in Fig. 11 [9].



*Fig. 11 : Interface for registration services [9]*

## **2. Middle-Tier Design**

The middle tier includes the MCCAS client application that has networking capabilities to interact with the client user interface and send commands to the MCCAS server. I have seen the similarity for the architecture between cloud space and traditional client/server structure. Indeed, the communication channel over cloud is based on HTTP request/response approach, mainly because the web app acts like an isolation layer with the hosting cloud system. The MCCAS application stores data in the cloud space where the data base store and provides SQL like query language for search, add, remove, and replace actions.

### **C. Deployment**

The prototype project took the best practice by using Eclipse IDE with plugins for MCCAS. In addition, we also use the Apache java library for Multipart Entity HTTP POST method support. The work on MACCAS is divided in modules/units and actual coding is started. Since, in this phase the code is produced so it is the main focus for the developer. To deploy the web app into the MCCAS cloud space, the developer needs to get a unique application ID from the MCCAS and apply it to the web app, and then click the “Deploy” button from the Eclipse IDE. The Eclipse will clean up the project and recompile it before delivering the output folder to the cloud.

**D. Testing**

Testing is a very important phase that evaluates and verifies that the functionality gives the desired results. Testing was done on multiple ways as following: -

- System Tests: after finishing the analysis and design phases, we start implemented the application, in each iteration a new version come out. In every version we test as a developer the function requirement of the system, check if all the processes work correctly, the flow of data in all the services are correct.
- Screen testing: Each screen tested with multiple screen size. After each screen developed was tested on all four screen sizes (small, medium, large, and x large).
- Integration testing: This test need in every iteration from the methodology, because a new features and functionalities add to the system, so we need to ensure that this functionality is running correctly and does not affect other functionalities.
- Acceptance testing: we present the application for 20 users, admission & registration staff, and IT staff, to measure the acceptance level. So we changed the design of the interfaces many times to be satisfied with the users.

**E. MCCAS Evaluation**

After finishing MCCAS development, we made a survey. This survey answered by a group of academic staff and students who's used the application and answered the survey questions. The survey answered by 220 students and 51 academic staff with different levels, gender and different phones types such as: Galaxy Tab, Galaxy S3, Galaxy S4, Galaxy Ace, Nokia Lumia 920, iPhone 4, iPhone 4S, iPhone 5, Sony Xperia z2, Sony Xperia m and HTC. The survey emphasize that MCCAS satisfies academic staff and student needs and run with high performance and compatible with all OS devices and achieve the main objective of our project. 89% of the students were in favor of the registration courses process were comfortable, 100% of the evaluators emphasize the view of table of exams and schedule courses were easy, 40 % said that transcript view in suitable way, 89% said it is easy to see the courses, also financial record was encouraged by 59%, the application provides academic plan 73% appropriately, but 60% support the electronic applications [10, 17] . Also I test the application performance by the bit-rate in 1 second, every activity how many time it take using MCCAS and in the tradition registration system for example to client need to register a course how many time need to finish this activity, on the other hand how many times needed in the tradition registration (Table 1. shows an example).

TABLE 1. PERFORMANCE - MCCAS VS. WEBSITE(TRADITION)

Functions	MCCAS (ms)	Website (ms)
Request Transcript	54	118
Register Courses	72	129
Request courses Schedule	38	103
Request Exams Schedule	36	98
Request Financial Report	63	122

**V. DISCUSSION**

Agile methodology was adapted to develop MCCAS application, it is incorporating the SDLC phases starting from the Planning, analysis, design, implementation, test and evaluation up to the Maintenance phase in each iteration. During the prototype development, we were benefited from the free “out of box” GAE admin console - it reduced the server side maintenance work. We also took advantage of the free quota of the GAE cloud resources (CPU and storage) throughout the entire development process and ended up with zero cost - this PaaS platform does helped us in making the best-ever cost saving on server holding. While we were focusing on the technical details with the prototype modeling and implementation, we do look at the mobile cloud impact to business model and management process. There is no doubt that mobile cloud computing opens a wide door enabling innovations. Using Agile let the Stakeholders more engaged with the application, also to be involved throughout building the application by each iteration the users deal with it. New features are delivered quickly and frequently, with a high level of predictability for the customers, also new or changed items can be planned for the next iteration, providing the opportunity to introduce changes within a few weeks. By allowing the client to determine the priority of features, so the application can serve the features that provide the most business value so focused on Business Value. By producing frequent builds and conducting testing and reviews during each iteration, quality is improved by finding and fixing defects quickly and identifying expectation mismatches early and let the application more easily for the users.

## VI. CONCLUSIONS

Even though Cloud Computing has been evolved for a couple of years, there are still fields need to be improved, especially in the Mobile Software industry, as the available APIs are still mainly targeting on desktop web based application clients. Several software engineering frameworks and methodologies have been developed to provide solutions for deploying cloud computing resources on mobile application development. Agile methodology is one of the most commonly used methodology in the field. This paper presents the MCCAS a Web and Mobile application that provide feature for the Palestinian higher education/academic institutions. An Agile methodology was used in the development of the MCCAS but in parallel with emphasis on Cloud computing resources deployment. MCCAS application developed using Agile methodology, it is incorporating the SDLC phases starting from the Planning, analysis, design, implementation, test and evaluation up to the Maintenance phase in each iteration. Also many related issues is discussed such as how software engineering modern methodologies (advances) influenced the development process. Using Agile let the Stakeholders more engagement with the application, also to be involved throughout building the application by each iteration the users deal with it. So the software engineering advances make revolution in mobile application. During the experience in this paper of adopting Agile software development practices on building mobile application, it was seen solutions delivered on time and with a higher degree of customer satisfaction. By incorporating the ability to change, it was able to better incorporate feedback from demos, usability testing, and client and customer feedback.

## REFERENCES

- [1.] Parameswari, M.R. and D.N. Prabakaran, An Android Enabled Mobile Cloud Framework for Development of Electronic Healthcare Monitoring System using VPN Connection. *International Journal*, 2013. 1(7).
- [2.] Khan, N., et al., Cloud Computing: Comparison of various features, in *Digital Enterprise and Information Systems*. 2011, Springer. p. 243-254.
- [3.] Mell, P. and T. Grance, *The NIST definition of cloud computing*. 2011.
- [4.] Dinh, H.T., et al., A survey of mobile cloud computing: architecture, applications, and approaches. *Wireless communications and mobile computing*, 2013. 13(18): p. 1587-1611.
- [5.] Khan, A.N., et al., Towards secure mobile cloud computing: A survey. *Future Generation Computer Systems*, 2013. 29(5): p. 1278-1299.
- [6.] Lynch, M., T. Cerqueus, and C. Thorpe. Testing a cloud application: IBM SmartCloud inotes: methodologies and tools. in *Proceedings of the 2013 International Workshop on Testing the Cloud*. 2013. ACM.
- [7.] Moe, N.B., T. Dingsøyr, and T. Dybå, A teamwork model for understanding an agile team: A case study of a Scrum project. *Information and Software Technology*, 2010. 52(5): p. 480-491.
- [8.] Zarinah, I.I.S.S.S. and M. Kasirun, AGILE-BASED SOFTWARE PRODUCT DEVELOPMENT USING CLOUD COMPUTING SERVICES: FINDINGS FROM A CASE STUDY.
- [9.] Naser, S.S.A., M.A. Ghosh, and R.R. Atallah, Mobile Cloud Computing: Academic Services for Palestinian Higher Education Institutions (MCCAS). *International Journal of Research in Engineering and Science (IJRES)*, 2015.
- [10.] Naser, S.S.A., M.A. Ghosh, and R.R. Atallah, Secure Mobile Cloud Computing for Sensitive Data: Teacher Services for Palestinian Higher Education Institutions. *International Journal of Advanced Science and Technology (IJAST)*, 2016.
- [11.] Abrahamsson, P., *Agile Software Development Methods: Review and Analysis (VTT publications)*. 2002.
- [12.] Uskov, V., et al. Engineering innovative Mobile Computing curriculum. in *Global Engineering Education Conference (EDUCON)*, 2013 IEEE. 2013. IEEE.
- [13.] Uskov, V.L. Mobile software engineering in mobile computing curriculum. in *Interdisciplinary Engineering Design Education Conference (IEDEC)*, 2013 3rd. 2013. IEEE.
- [14.] Almudarra, F. and B. Qureshi, Issues in Adopting Agile Development Principles for Mobile Cloud Computing Applications. *Procedia Computer Science*, 2015. 52: p. 1133-1140.
- [15.] Kalathoti, K. and M.K. Kumar, A Case Study on implementation Mobile Software Engineering in Cloud Environment. 2014.
- [16.] Yu, W.D. and H. Yuan. An approach to explore mobile software engineering advances in cloud computing environment. in *Computer Software and Applications Conference Workshops (COMPSACW)*, 2011 IEEE 35th Annual. 2011. IEEE.
- [17.] Naser, S.S.A., M.A. Ghosh, and R.R. Atallah, Using Social network in Higher Education A Case Study on the University of Palestine. *International Journal of Engineering Research and Applications (IJERA)*, 2014.
- [18.] Barhoom, T.S., M.A. Ghosh, Reduce Resources for Privacy in Mobile Cloud Computing Using Blowfish and DSA Algorithms. *International Journal of Research in Engineering and Science (IJRES)*, 2016.