



ASHESI UNIVERSITY

**Axis Mobile: A Mobile Application to Support the  
Processes of Axis Human Capital Limited, a Human  
Resource Management Company in Ghana**

**Applied Project**

B.Sc. Computer Science

**Okoh-Asirifi Elvis**

**2019**

# **ASHESI UNIVERSITY**

## **Axis Mobile: A Mobile Application to Support the Processes of Axis Human Capital Limited, a Human Resource Management Company in Ghana**

### **APPLIED PROJECT**

Applied Project submitted to the Department of Computer Science, Ashesi  
University in partial fulfilment of the requirements for the award of  
Bachelor of Science degree in Computer Science.

**Elvis Okoh-Asirifi**

**April 2019**

## DECLARATION

I hereby declare that this applied project is the result of my original work and that no part of it has been presented for another degree in this university or elsewhere.

Candidate's Signature:

.....

Candidate's Name:

.....

Date:

.....

I hereby declare that the preparation and presentation of this applied project were supervised in accordance with the guidelines on supervision of applied project laid down by Ashesi University.

Supervisor's Signature:

.....

Supervisor's Name:

.....

Date:

.....

## **Acknowledgement**

I would like to express my sincere gratitude to God for His wisdom and strength that helped me complete this project successfully. I would also like to thank my supervisor, Dr Ayorkor Korsah for her time, feedback and critiques. I would like to appreciate Dr Esi Ansah, Chief Executive Officer of Axis Human Capital Limited, and all the employees and clients of Axis for helping me complete this project successfully. Finally, I would like to express my gratitude to my friends who allowed me to borrow their devices to run tests for this application. I thank them for their honest feedback and critiques too.

## **Abstract**

The increasing accessibility to internet resources has made several companies adopt the use of such resources in their processes. Axis Human Capital Limited, a human capital management and development company in Ghana, needed a mobile application that mitigates the problems that their clients face when using their website and has other important features that successfully support their processes and improve them where necessary. This paper proposes a hybrid mobile application, Axis Mobile, and a web application, Axis Notification Center, that meet Axis' needs. The proposed solutions address the limitations of Axis's website and provides important features that complement it. Axis Mobile and Axis Notification Center make Axis' clients much happier and Axis' processes much faster.

## Table of Contents

DECLARATION .....	iii
Acknowledgement .....	iv
Abstract .....	v
Table of Contents .....	vi
List of tables .....	ix
List of figures.....	x
Chapter 1: Introduction .....	1
Chapter 2: Requirement Analysis .....	6
2.1 Project Perspective.....	6
2.2 Functional Requirements for Axis Mobile.....	6
2.3 Use Case Diagram for Axis Mobile.....	8
2.4 Functional Requirements for Axis Notification Center .....	9
2.5 Use Case Diagram for Axis Notification Center .....	9
2.6 Domain Requirements .....	9
2.7 Non-functional Requirements for Axis Mobile.....	10
2.8 Non-functional Requirements for Axis Notification Center .....	10
2.9 Usability Requirements of Axis Mobile and Axis Notification Center.....	10
Chapter 3: Architectural Design .....	11
3.1 Client-Server Architecture .....	12

3.2 Model-View-Controller .....	13
3.3 Architecture of Axis Mobile .....	15
3.4 Architecture of Axis Notification Center.....	17
3.5 Database Design of Axis Mobile.....	18
3.6 Database design of Axis Notification Center .....	19
Chapter 4: Implementation .....	21
4.1 Technology Stack for Axis Mobile.....	21
4.2 Framework for Axis Mobile.....	24
4.3 Programming Languages for Axis Mobile.....	24
4.4 Technologies for Axis Mobile.....	25
4.4.1 Ionic Framework .....	25
4.4.2 PHP.....	25
4.4.3 phpMyAdmin .....	26
4.4.4 OneSignal .....	26
4.4.5 SQLite.....	26
4.5 Axis Mobile .....	26
4.6 Choice of Programming Languages for Axis Notification Center .....	31
4.7 Choice of Framework for Axis Notification Center .....	31
4.8 Axis Notification Center .....	32
Chapter 5: Testing and Results .....	38

5.1 Development Testing .....	38
5.1.1 Unit testing.....	38
5.1.2 Component testing.....	39
5.1.3 System Testing .....	41
5.2 Release Testing and User Testing .....	41
5.2.1 Beta testing .....	42
Chapter 6: Conclusion.....	46
6.1 How Axis Mobile will support Axis' processs .....	46
6.2 How Axis Mobile will enhance users' experience with Axis Human Capital.....	47
Recommendations for future work.....	48
References .....	50
Appendix .....	51
A.1 Interview Questions .....	51
A.1.1 Employees of Axis: .....	51
A.1.2 Job seekers: .....	51
A.1.3 Employers: .....	51
A.2 Screenshots of Axis Mobile .....	52

## **List of Tables**

Table 3.1: Table 3.1 Database tables used by Axis Mobile and Axis Notification Center.....	18
Table 4.1: A comparison of native, hybrid and cross-platform applications.....	21

## **List of Figures**

Figure 2.1: Use Case Diagram for Axis Mobile.....	8
Figure 2.2: Use case diagram for Axis Notification Center.....	9
Figure 3.1: Model-View-Controller Architecture of Axis Mobile.....	15
Figure 3.2: Client-Server Architecture for Axis Mobile.....	16
Figure 3.3: Client-Server Architecture for Axis Notification Center.....	17
Figure 3.4: Entity Relationship Diagram for Axis Notification Center.....	20
Figure 4.1: Axis Mobile Log in page.....	27
Figure 4.2 Forgot (or resetting) password interface.....	28
Figure 4.3: Home page after a successful login (for a job seeker).....	29
Figure 4.4: Home page after a successful login (for an employer).....	30
Figure 4.5: Login Page for Axis Notification Center.....	32
Figure 4.6: Home Page of Axis Notification Center.....	33
Figure 4.7: Main Page (scrolled down) of Axis Notification Center.....	34
Figure 4.8: Sending a Phone Notification to Elvis Okoh-Asirifi.....	35
Figure 4.9: Phone Notification from Axis Notification Center.....	36
Figure 4.10: Sending Bulk Notifications.....	37
Figure 5.1: Figure 5.1: Job Status (a) and Job Details (b) pages .....	40
Figure 5.2: Login and Home pages of the first version of Axis Mobile.....	42

Figure 5.3: Login (a) and Home (b) pages.....	43
Figure 6.1: Search results.....	48

## **Chapter 1: Introduction**

Axis Human Capital Limited is a human resource management and development company in Ghana. They offer recruitment and selection services for employers; they help them find a good candidate for a job opening. Axis posts the job opening on their website so that job seekers who see them can apply with their cover letters. Axis also offers advisory services and technical support for companies in the areas of branding strategies, stakeholder management, workspace design, talent management, etc. They also organise training programs for employees at the request of their employers. Axis has built a reputation for “recruiting high-calibre candidates for employers, and working with them to develop their organizational capacity through advisory services and corporate training” [7]. Axis has worked with companies such as Ashesi University, First National Bank, Calbank, Tullow Oil, Kosmos Energy, etc. since its inception [7].

Axis works with job seekers and helps them "map out their career paths and prepare them for career opportunities through individual career coaching, workshops and CV reviews" [7].

The increasing access to the resources provided by the Internet has caused several organisations to adopt its use in their human resource management processes [1]. In the recruitment of job seekers for a vacant position, the use of the Internet (e-recruitment) to support the recruitment process hastens the tasks of processing applications and communicating with candidates [3]. Kesen[5] also noted that making use of the Internet and related applications make the human resource management process much easier, faster and more dependable.

To make their interaction with clients much better, Axis has developed a website which facilitates their interaction with their clients (employers and job seekers). The website provides information about the services the company provides and various job openings. The website also

has a reading lounge where people can find articles about human resource development and how to prepare oneself for the professional world.

Employers can use the website to request for recruitment services from Axis by providing all the details of the job such as the job title and description, the minimum education required, the nature of the job and the industry. This strategy has been beneficial and has made the process of outsourcing some of the human resource management functions much easier for employers [3].

Outsourcing some of the human resource functions helps managers cope with competing demands and time-sensitive issues such as recruiting an applicant from a pool of over fifty applicants [2].

Axis requires job seekers who seek to apply for jobs listed on their website to first sign up and upload a resume. After that, the job seeker can easily log onto the web-based application and browse through all the available job vacancies, view the qualifications needed to apply for each of them and if they meet the requirements, apply by submitting their cover letters (using the web-based application).

Even though the web application makes it easier for job seekers to have access to relevant information about job vacancies, the job seekers are required to regularly check the company's website to access information about current job vacancies, training podcasts and videos and other general information. The website is also the only means by which they can track the status of their applications. The responses obtained from interviewing job seekers that work with Axis showed that a problem they (job seekers) often encounter is that they miss some information because they fail to check updates on the website.

Another problem with the website is that a job seeker can view all the jobs he has applied to, but he cannot view the status of his applications. That is, he cannot determine if he was selected or rejected or if his application is still pending. For most of the job seekers that were interviewed, that was one of their primary concerns.

Several job seekers also mentioned that they find it very hard to reset their passwords on the Axis website when they forget their old passwords. They complained that they often had to call employees of Axis to help them reset their passwords, which could be daunting sometimes.

Others explained that they send emails to Axis, but their emails are ignored so they are compelled to go to Axis' office to get them reset. Others, out of frustration and desperation, create new accounts. Employees of Axis mentioned that a few of the job seekers they work with have multiple accounts and at some point, they had to delete some of the accounts.

While difficulty in resetting passwords was a primary concern to some job seekers, others still complained about the login details they provide when logging into Axis' website. Users of the system explained that they provide a least one email address when signing up however the email seems to be useless when signing in to Axis' website (even though they can sign in to other web services using their email addresses). They mentioned that the six-digit code (Axis ID) that is generated for them when they sign up is very easy to forget and hence, they would prefer a system that would enable them to log in with either their email addresses or Axis IDs.

Axis also noted that employers have expressed little interest in using the website to access their services. An interview with the employees of Axis revealed that most employers would rather call them on phone and request their services than access those services through the website. They noted that perhaps, the number of fields or the amount of information employers would

have to provide on the website is a lot, and that scares them away. They, therefore, wished for a system that would require little but adequate information from employers who wish to access Axis' services, especially for recruitment and selection services.

Axis needed a mobile application in addition to their web-based application to help them resolve these issues. The company wanted job seekers to be able to receive notifications on all job vacancies that are posted on the website. The management also wanted the users to be able to know the status of their job applications much faster. Thus, whether users' applications move to the next level or they are rejected, they should be notified quickly through the application. The application should also enable job seekers to search for jobs that match their interests and qualifications.

Axis required that the mobile application should help them reduce the number of instances of multiple accounts created by job seekers and should make it easier to change and reset passwords without involving employees of Axis.

To further foster a more vibrant and faster interaction with clients, Axis wanted the mobile application to have a chat feature with which clients can make complaints and give feedback quickly.

Software applications such as BambooHR [8], Kronos Workforce Ready [9], Zoho Recruit [10] and iCIMS Talent Acquisition Suite [11] have all been developed to make the hiring process much easier and faster. These applications indeed facilitate the hiring process as they allow the companies to track job applicants, manage and monitor existing employees' performance, prepare payroll, and keep track of attendance at work [8,9,10,11]. However, they do not cater for all of Axis' needs. These applications are tailored towards the needs of recruiters, rather than

those of job seekers. For instance, these software packages do not allow job seekers to book sessions with Axis to help them review their resumes and they do not allow clients to have a live chat with Axis' employees in case they want to offer feedback or ask for more clarification.

Axis mobile is a mobile application that has been developed to help Axis resolve all the problems they face with their current system. The mobile application has a chat feature and allows clients to chat with employees of Axis at a very low cost. It also enables job seekers to receive notifications on job vacancies, training videos and podcasts and other general information that the company posts on its website. It makes it easier for users to track their job applications and know the status of each job application without logging into the company's website. The issue of multiple accounts has been tackled from the root by allowing users to sign in using either their emails or Axis IDs and by allowing users to easily change or reset their passwords when they forget without sending an email to employees of Axis or making a phone call to them.

The simple interface and little information required from employers who wish to request recruitment services from Axis have improved employers desire to access Axis' services through the mobile application.

Axis Mobile also comes with a companion web application called Axis Notification Center. Axis Notification Center enables employees of Axis to send emails and phone notifications to their clients. Axis Notification Center integrates into Axis' administrators' website and allows emails and phone notifications to be sent to all customers at once or to particular segments (individual or group). This web application primarily facilitates the sending of custom notifications to clients that have downloaded and installed Axis Mobile. However, it also allows emails to be sent to all clients of Axis without using an email software such as Microsoft Outlook.

## **Chapter 2: Requirement Analysis**

This chapter describes the requirements for the mobile application that has been built for Axis Human Capital Limited (Axis Mobile) and the companion web application. The chapter includes the functional, non-functional, domain and usability requirements. Face to face interviews were conducted with employees of Axis, as well as the job seekers and employers that work with Axis. The results from the interviews inspired the requirements for the application. The questions that were asked during the interviews have been stated in the appendix.

### **2.1 Project Perspective**

Axis Mobile is a mobile application for use on mobile devices – smartphones and tablets. The application supports both the Android and IOS platforms. The application requires an internet connection to handle users' requests and to show various job vacancies posted by Axis and the status of job applications. The application integrates into the database that the company's website uses. Thus, Axis Mobile does not have a separate database. It however caches some of the user's data on the device for quick accesses.

Axis Notification Center is a web application that can be accessed anywhere by employees of Axis. The application supports all browsers. The application requires active internet connectivity to send notifications and emails to clients. The application also integrates into the database that the company's website uses. Thus, Axis Notification Center does not have a separate database. It is rather an add-on to Axis' website.

### **2.2 Functional Requirements for Axis Mobile**

Users of the application (job seekers and employers) should log in with their Axis IDs or email addresses and their passwords. They should not be able to sign up using the mobile application

due the amount of data they need to provide when signing up. The sign-up process should be done on the company's website, then users would use their credentials to log in to the mobile application. However, if they forget their passwords, they should be able to reset it using the application.

The application should allow job seekers to receive notifications on job vacancies and general information that would be posted on the Axis website. Employers should also be able to receive notifications on the status of the jobs they posted. Job seekers should be able to select the kinds of notifications to receive. Job seekers should also be able to use the application to track their job applications and know the status of each job application. The application should also have a live chat feature that would allow Axis' clients (both employers and job seekers) to chat with Axis.

Job seekers should be able to place phone calls to Axis right from the mobile application.

Articles posted by Axis on their website should be available on the mobile application so that both employers and job seekers can read them.

Job seekers should also be able submit their sample work documents using the mobile application. A sample work document is a document that gives a detailed description of a job that someone worked on. It explains the nature of the job, the skills that were applied on the job and the person's achievements while working on the job.

The application should allow a job seeker to book resume-editing sessions with Axis. This will help track how many people will visit Axis for resume editing on a particular day.

Employers should also be able to track the status of their recruitment process that is being handled by Axis. They should be allowed to post new job vacancies using the mobile application.

Answers to frequently asked questions about Axis' services should be on the application for both job seekers and employers to view. This feature can also limit the number of calls clients make to Axis every day. The use case diagram for Axis Mobile, which is a summary of the functional requirements, is shown in figure 2.2 below.

### 2.3 Use Case Diagram for Axis Mobile

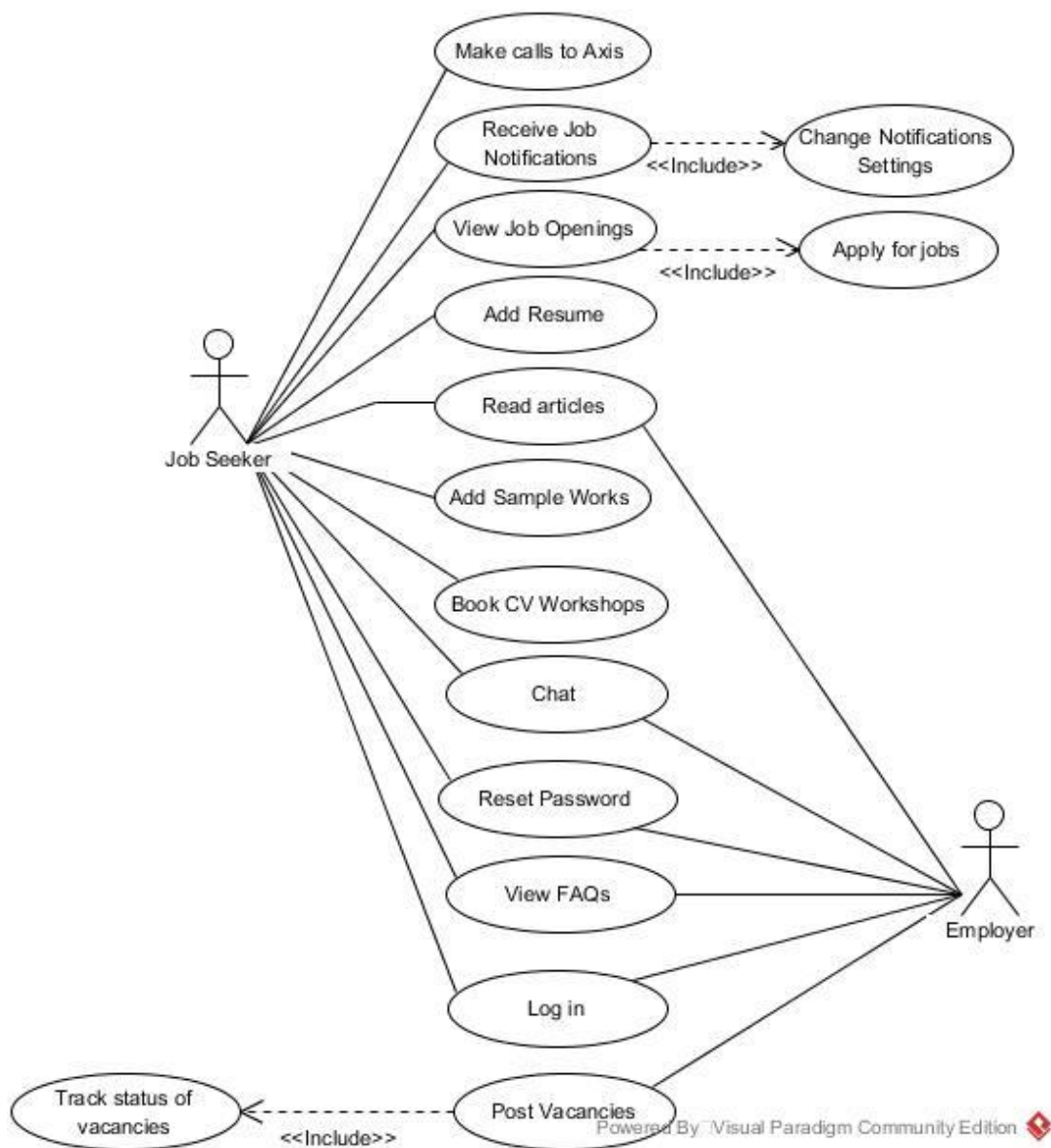


Figure 2.1: Use Case Diagram for Axis Mobile

## 2.4 Functional Requirements for Axis Notification Center

The application should allow employees of Axis to log in with their Axis IDs or email addresses and their passwords. The application should allow the employees to send notifications and emails to individual clients and groups of clients. The application should also keep track of all notifications and emails that have been sent out to clients.

## 2.5 Use Case Diagram for Axis Notification Center

The use case diagram for Axis Notification Center, which summarizes the functional requirements of the application, is shown below.

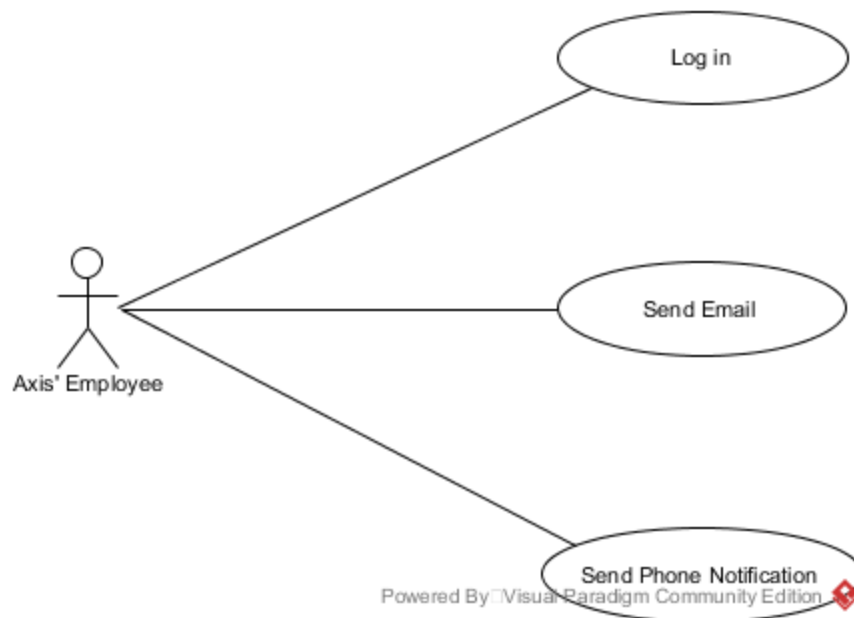


Figure 2.2: Use case diagram for Axis Notification Center

## 2.6 Domain Requirements

Axis Mobile should support Android and IOS devices. Axis Notification Center should support all browsers.

## **2.7 Non-functional Requirements for Axis Mobile**

Axis Mobile must be very fast in terms of performance. Making requests and receiving notifications should be fast and efficient. The application must be secure, so that hackers cannot penetrate the system and gain access to confidential information. The system should also be modular to allow more features to be added and allow outdated features to be easily updated. The application should be accessible to all clients who wish to use it, once they have access to internet connection. The application should also be scalable to accommodate Axis' increasing customer base. Thus, the application should not become slow as the number of users increases. Users of the application should authenticate themselves using their Axis IDs or email addresses and their passwords.

## **2.8 Non-functional Requirements for Axis Notification Center**

The application must be very fast in terms of performance. Sending notifications and emails should be fast and efficient. The application must be secure, so that hackers cannot penetrate the system and gain access to confidential information. The system should also be modular to allow more features to be added and allow outdated features to be easily updated. The application should be accessible to all employees who wish to use it, once they have access to internet connection and should allow them to be authenticated using their emails addresses or Axis IDs and passwords.

## **2.9 Usability Requirements of Axis Mobile and Axis Notification Center**

The applications should be very easy to use, and the icons should not be confusing. After a few minutes of first-time usage, the average user should be able to navigate the applications with ease.

### **Chapter 3: Architectural Design**

Ian Sommerville [6:148] explains architectural design as “understanding how a system should be organized and designing the overall structure of that system.” This chapter describes the overall structure of Axis Mobile, as well as the strengths and weaknesses of the chosen architecture.

The choices of the architecture of both Axis Mobile and Axis Notification Center were influenced by the non-functional requirements of the two systems – performance, security, modularity, scalability, safety, maintainability and availability. Sommerville [6] noted that there exists a strong relationship between the architecture of the system and the non-functional requirements and that the architectural style of a system should be determined by the non-functional requirements of the system that is being developed. The considerations that were made before the architectures of the system were determined include:

- **Performance:** The application should be able to handle user requests very quickly and provide feedback almost immediately. Therefore, the chosen architecture should be one that allows for quick processing of user requests.
- **Security and safety:** The application should be secure from illegal access and database injections so that hackers cannot penetrate the system and gain access to confidential information. The chosen architecture should, therefore, ensure that the data is kept safe from illegal accesses and corruption.
- **Modularity and maintainability:** The code used to develop the application should be modular, easy to maintain, easy to reuse and easy to add on. Thus, the chosen architecture should support code reuse and maintenance.

- **Availability:** The application should be available for use at every point in time, as long as the user has access to the Internet. Hence, the architecture should support the availability of the system at all times.
- **Scalability:** The application should not break or run slowly as the number of users increases. Hence, the chosen architecture should support scalability when the number of users increases.

Considering the factors mentioned above, as well as the nature of the applications, Model-View-Controller and Client-Server Architecture were the most suitable architectural design patterns to use for Axis Mobile while the Client-Server Architecture was the most suitable for Axis Notification Center. Sommerville [6] also mentioned that Model-View-Controller and Client-Server Architecture are more suitable for web applications and applications that involve a shared database that will be accessed from various locations.

### **3.1 Client-Server Architecture**

According to Sommerville [6], client-server architecture is organized into a set of services that are delivered by different servers. The clients can then access these services from the various servers.

In the case of Axis Mobile, the services are the functionality offered by the application, such as the ability to reset passwords, ability to view job vacancies, ability to view the status of job applications etc. and the clients are the job seekers or employers who use their smartphones to access the services. Clients access the services provided by a server by making requests to the server. These requests were implemented when developing the application. For instance, when a job seeker wishes to view all job vacancies posted by Axis, they click on the vacancies button on the home page of the application. This click action sends an HTTP request from the application

to the server. The server, which is hosted on the Internet, receives the request, processes it and sends a JSON response to the client (mobile application). The application decodes the response and displays it in a nice format on the vacancies page for the job seeker.

In the case of Axis Notification Center, the services are the email services provided by the remote server on which the application is hosted, and the notifications services provided by the notification service provider which the application is connected to.

Client-server architecture is often regarded as an architectural pattern that involves multiple servers. However, the independent services can also be implemented on a single server [6].

Client-server architecture allows for separation and independence. In the case of Axis Mobile, the client, which is the application installed on the phone, is separate and independent from the server, which is hosted remotely. In the case of Axis Notification Center, the client is the web application. The client communicates with the server, which is hosted remotely, by sending requests to send emails or notifications to clients of Axis. Hence, additional features can be added to the application without affecting the server and changes can also be made on the server side without affecting the application itself. This advantage makes it possible for modularity, code reuse and code maintenance.

A downside to the client-server architecture is that performance depends on the network strength, so performance is not predictable.

### **3.2 Model-View-Controller**

Model-view-controller separates the presentation (view) from the data. The architecture divides the system into three different components – model, view and controller, that interact with each other. The model component manages the data the clients will be accessing (the central

database) and the operations that will be performed on the data. The view defines the interface that the user sees (the front end). The controller component manages users' interactions on the view (such as mouse clicks, swipes, keyboard inputs, etc.) and passes the interactions to the view and model for processing.

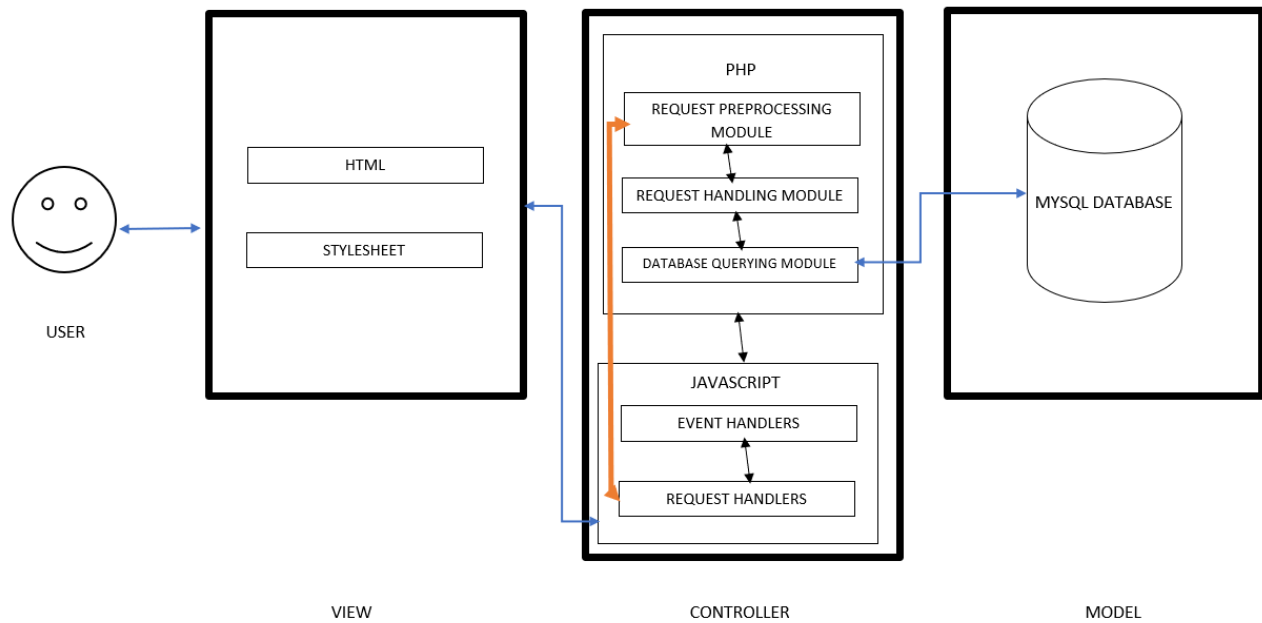
In the case of Axis Mobile, the view refers to the pages that will be viewed by the users of the application. The controller refers to the code that reacts to users' interaction on the views such as user clicks, submission of forms etc. The model is the database that keeps users' information.

Model-view-controller is used when there are multiple ways the data can be viewed and interacted with [6]. The advantage of model-view-controller is that it allows the data (model) to change without affecting the view. Likewise, the view can also be changed without changing the data. Sommerville [6] adds that model-view-controller "supports presentation of the same data in different ways with changes made in one representation shown in all of them." Hence, model-view-controller helps ensure maintainability of code, scalability of the entire system, safety of the data and availability of the system.

The main disadvantage of model-view-controller is that it may require additional code and code complexity, even though the interactions between the data and the models are simple [6].

For better performance of the system, both the client-server architecture and the model-view-controller architecture were used to develop Axis Mobile. Axis Mobile harnessed the advantages of both patterns to ensure that it delivers the best experience for users (performance and availability) while ensuring that the data stored is secure and the code can be easily maintained and reused. The entire system can also be easily scaled to accommodate the growing number of users without running slowly.

### 3.3 Architecture of Axis Mobile



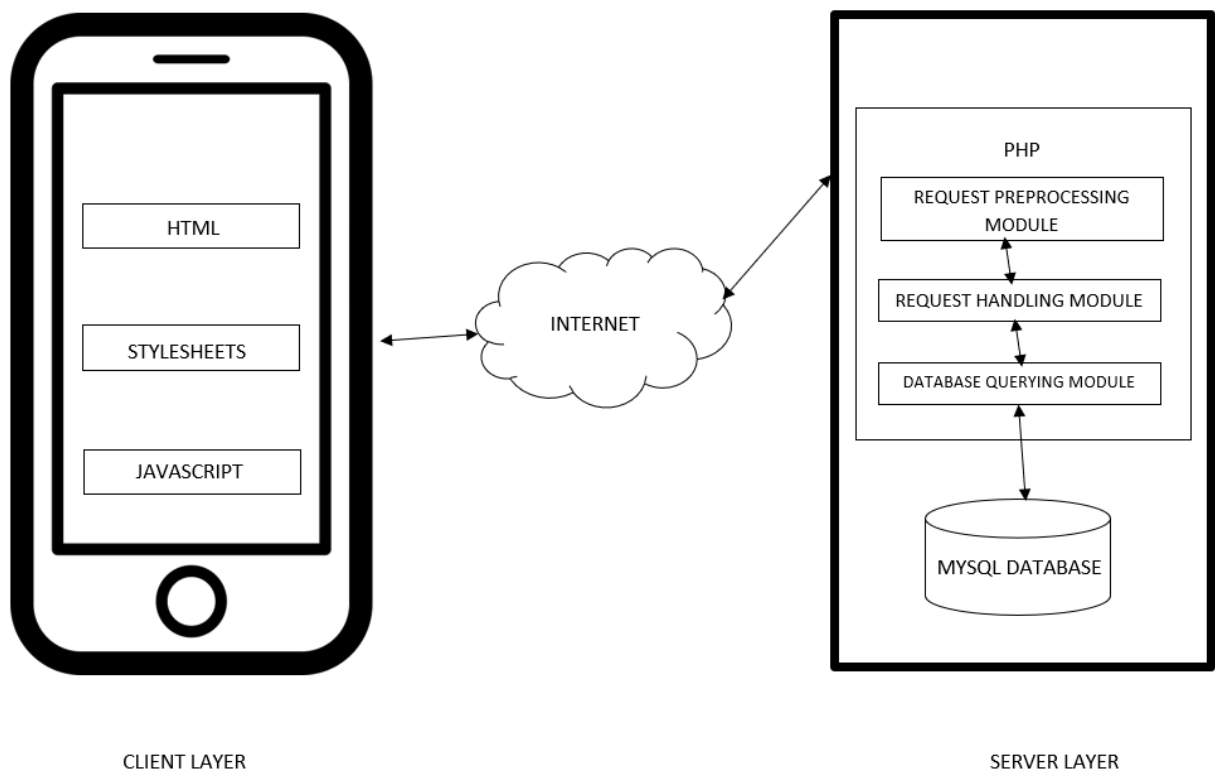
*Figure 3.1: Model-View-Controller Architecture of Axis Mobile*

The user sees the views. His interactions on the views such as filling of forms, uploading of files and clicking on buttons are all handled by the controller. The controller manipulates the model (inserts into, updates or deletes from the model) based on the user's interactions. The response from the model is also passed to the views through the controllers. The views are updated for the user to see the response of his interactions.

For instance, if a user wants to upload a resume, he first locates the resume from the phone's storage, enters a name for the resume and clicks the submit button (all these actions are made on the view). The clicking of the submit button is handled by the JavaScript controller, which sends the form data to the PHP script on a remote server. The PHP script processes the request and stores the user's resume details in the database (model). On a successful insertion, a message is

sent back from the PHP script to the JavaScript script which updates the view by alerting the user of a successful insertion.

The model is hosted on a remote server. The interactions of the user are sent as requests to the model, which are processed, and appropriate responses sent back and updated on the view. These set of actions illustrate the client-server architecture model of Axis Mobile. Figure 3.2 shows the client-server architecture of Axis Mobile.



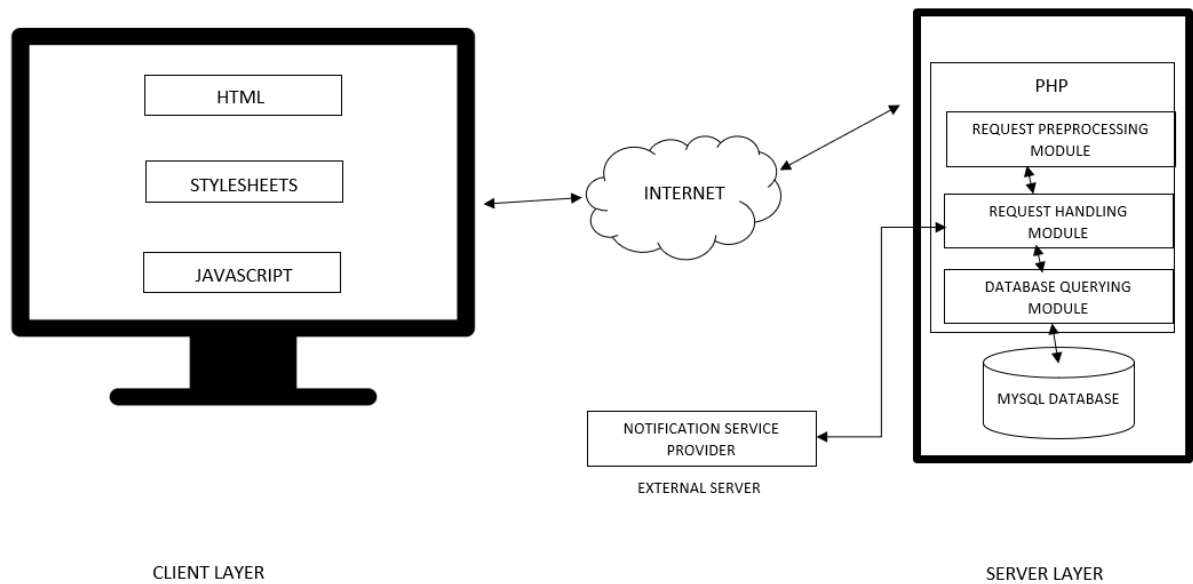
*Figure 3.2: Client-Server Architecture for Axis Mobile*

The users, through their interactions with the views, will send requests to the controller, which will in turn manipulate the model and send the response from the model back to be displayed on the views. In the case of Axis Mobile, a part of the controller (PHP scripts) and the model

(MySQL database) both reside on the remote server. This allows multiple users across the world to access Axis' central database of clients' records, job postings etc.

### 3.4 Architecture of Axis Notification Center

Figure 3.3 shows the architecture of Axis Notification Center.



*Figure 3.3: Client-Server Architecture for Axis Notification Center*

Users of the application send requests to the servers, depending on which action they want to perform. When the user wants to send an email, a request is made to the email server to send an email. On the other hand, when a notification needs to be sent, a request is made to the notification server which is managed by the notification service provider and the request is processed. The result is that the notification is sent to the target audience and a response is sent back to the user who requested the notification service.

### 3.5 Database Design of Axis Mobile

Axis' database consists of about fifty-seven (57) tables. However, Axis Mobile uses only eighteen (18) of them. The table (Table 3.1) below shows that tables used by Axis Mobile and a description of each table.

Table 3.1 Database tables used by Axis Mobile

Table	Description
Axis_clients	It keeps information of all the companies Axis works with. It keeps the name of each company, the website of the companies and the contact information of the companies.
Axis_company_categories	It stores the nature of the services of the companies Axis works with. It stores the IDs and the nature of business for each of the companies.
Axis_critiquing	It stores the details of customers that book resume-editing sessions with Axis. It stores their names, phone numbers, session times and dates.
Axis_devices	It stores the IDs of the phones used by those who have installed Axis Mobile. This information helps to send specific notifications to users of those devices. The columns of the table are the AxisID of the user and the phone's identification number.
Axis_educationlevels	It stores the various educational levels that may be required by a job.
Axis_employers	It stores information relating to employers only, such as which companies they work in and their role in the company (an employer can be the Human Resource Manager of a company. Axis categorizes them as employers).
Axis_industry	It stores the industries which the various companies that work with Axis belong to. It stores the industry name and the company's ID.
Axis_jobs	It stores all the details of a job such as the job titles, job description, minimum qualification, salary etc.
Axis_jobseeker	It stores information relating to only job seekers such as their mailing addresses, birthdays, star ratings etc.

Table	Description
Axis_jobseeker_coverletters	It stores information about the cover letters of the job seekers. It stores the names of the cover letters, the AxisIDs of the owners of the cover letters and the location of the cover letters in Axis' file system.
Axis_jobseeker_job_applications	It stores the information of the jobs that job seekers have applied to. It stores the status of the applications and the application dates.
Axis_jobseeker_resumes	It stores information about the resumes of the job seekers. It stores the names of the resumes, the AxisIDs of the owners of the resumes and the location of the resumes in Axis' file system.
Axis_jobseeker_sampleworks	It stores information about the sample work documents of the job seekers. It stores the names of the documents, the AxisIDs of the owners of the documents and the location of the documents in Axis' file system.
Axis_jobstatus	It stores the set of possible job statuses. For instance, a job can be open to new applicants, closed to new applicants or on hold.
Axis_jobtype	It stores the set of possible job types such as full-time, internship and volunteer.
Axis_job_functions	It stores the functions involved in a job such as sales or retail, security, project management and design.
Axis_users	It stores the information of the users of Axis' system such as employees of Axis, job seekers and employers.
notifications	It stores all the notifications that have been sent to clients. It stores the date and time the notification was sent, the subject of the notification and the content (message) of the notification.

### 3.6 Database design of Axis Notification Center

Axis Notification Center also makes use of three out of the fifty-seven tables. The tables include the notifications table, the Axis\_devices table and the Axis\_users table. The Entity Relationship Diagram for the tables is shown in figure 3.4 below.

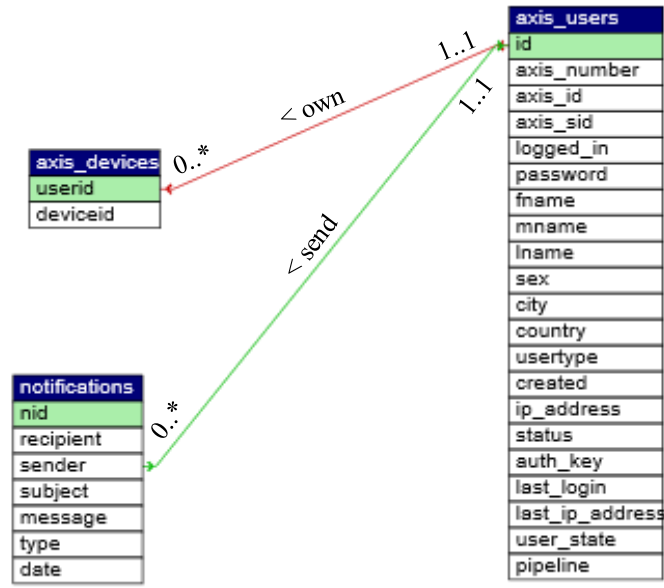


Figure 3.4: Entity Relationship Diagram for Axis Notification Center

The relationships between the tables show that a user (employee of Axis) can send zero, one or more notifications to other users, and a user can own zero, one or more devices.

## **Chapter 4: Implementation**

Several decisions were made to come up with a mobile application that satisfies Axis' needs and a web application that enables them to quickly send emails and phone notifications. Factors such as the technology stack, the framework and programming languages were needed to build the applications were considered in detail.

### **4.1 Technology Stack for Axis Mobile**

The first decision that had to be made was whether to build native applications for both the Android and IOS, build a cross platform application that would be built to support both Android and IOS or develop a hybrid application that would support both platforms. A comparison was made between the three technology stacks available for developing mobile applications. The comparison is summarized in the table below:

Table 4.1: A comparison of native, hybrid and cross-platform applications

	Native	Cross Platform	Hybrid
Description	In this approach, the application is built with a specific platform in mind. The application is developed using a programming language supported by the platform. Because the application is platform specific, the application has access to all the device features and has native look and feel.	In this approach, the application is developed for one platform, then it is 'converted' to other platforms. For instance, an application developed in Xamarin will have Windows as the target platform, but the application can be converted into an Android or IOS application. This makes the application work better in the target platform for which it is developed but experience performance lags or delays in the converted platforms.	In this approach, the application is developed with all platforms in mind. The application is developed to run within a browser, even though it is not noticeable when using the application. When development is complete, the application is built for the intended platforms on which the programmer(s) want it to work. Hence, it may not have native look and feel. However, Ionic 4 presents beautiful themes that mitigate this problem
Programming languages	Objective-C or Swift for IOS and Java or Kotlin for Android	Xamarin + C# or Titanium	HTML, CSS and JavaScript
Device Integration	Full device integration	Limited device feature support	Limited access to device features and platform-specific features.
Performance	Fastest performance	Performance delays	Slower interactions but are not so noticeable in Ionic 4
Responsiveness	Most responsive	Better responsiveness on target platform but has performance shortfalls on 'converted' platforms	Slower interactions but are not so noticeable in Ionic 4

	Native	Cross Platform	Hybrid
Cost	Most expensive to use especially when you want to develop for all platforms separately. Costs may include cost of new hardware and cost of hiring developers for the different platforms	It is relatively less expensive to build compared to native applications. However, development frameworks such as Xamarin are not free to use.	Very inexpensive to build as some of the frameworks are less expensive or open source.
Reusability of code	Code reuse is not possible because the different platforms use different programming languages.	Code can be reused easily because the application is developed using a single language for all platforms.	Code reuse is possible.
User experience	Provides the best user experience.	Provides the best user experience for the target platform.	Does not provide the best user experience to the user because of performance lags and slower interactions. However, for data retrieval applications with limited use of device resources, they work very well
Main advantages	Full device integration, good user experience, fast performance and high responsiveness	Less expensive, compared to native applications, run on multiple platforms and support code reuse.	Inexpensive to build, run on multiple platforms and easy to build and maintain.
Main disadvantages	Costs a lot to develop, in terms of money and time since the developers must develop for the platforms separately. Native applications can run on only one platform and requires more developers if deadlines are tight.	Limited support for device features, performance delays and poor user experience.	Limited access to device features, performance delays and poor user experience.

Axis Mobile needed to be developed very quickly to mitigate the problems that job seekers face with the existing system. Axis was also not ready to invest so much in the development of the application. Hence, native applications were not the immediate choice. A hybrid application was chosen because it is easier to build and maintain compared to a cross-platform application. A good framework such as Ionic 4 also provides beautiful themes that enhance the user experience of the application. Besides, certain functionality in cross-platform applications are not free to use so a hybrid application seemed to be the best choice.

#### **4.2 Framework for Axis Mobile**

A framework is a set of tools or resources that are used to build or manage a web application or a web service. After the choice of technology stack was made, the next thing that had to be considered was the choice of framework. Under hybrid application development, there are a number of frameworks that can be used. The Ionic framework was chosen because it has a great community of developers who are willing to help when developers run into problems while using the platform. It also supports push notifications (one of the features that Axis desires to have in the application) and has a less steep learning curve when the developer is familiar with HTML, CSS and JavaScript. A push notification is a short message or alert that can be sent to devices to inform or notify the users of the devices about something.

#### **4.3 Programming Languages for Axis Mobile**

The Ionic framework is built on HTML, CSS, JavaScript (Angular). However, since the application would integrate with the existing system (with a MySQL database and a PHP backend), the PHP language was also used to process requests sent from the application.

## **4.4 Technologies for Axis Mobile**

### **4.4.1 Ionic Framework**

Ionic framework is an open-source application development framework for building quality mobile and desktop applications using HTML, CSS and JavaScript [4]. Ionic is easy to learn and integrates very well with other frameworks [4]. It also supports various APIs (Application Programming Interfaces) such as OneSignal for push notifications. Ionic is integrated with Angular currently, but support for other JavaScript frameworks such as Vue and React are in development [4]. Angular, Vue and React are JavaScript frameworks that make building web applications much faster. Ionic uses the ‘write once, run anywhere’ approach to mobile development – where the programmer develops a single code base that runs on multiple platforms [4]. Ionic is also built on HTML, CSS and JavaScript, which are reliable and standardised web technologies. Hence, ionic applications can run on multiple platforms. The framework also provides pre-designed components and beautiful themes to enhance the user interface of the application on all platforms. Ionic also has a community of developers who are ready to answer questions and fix bugs that users of the framework may have or encounter [4].

### **4.4.2 PHP**

PHP stands for ‘PHP: Hypertext Preprocessor’ and it is a server-side language for developing web applications and creating dynamic web applications [12]. PHP was used to develop the backend for Axis Mobile because the current system that Axis uses was built using WordPress, which is also built on PHP. It was, therefore, easier to integrate the mobile application into the existing system because they were both running on the same backend scripting language.

#### **4.4.3 phpMyAdmin**

phpMyAdmin is an application written using PHP that handles the administration of MySQL over the web [13]. The Axis web application uses phpMyAdmin to manage the MySQL database that the company uses to store information about its clients.

#### **4.4.4 OneSignal**

OneSignal is a high-volume push notification service for both web and mobile applications [14]. This service was combined with PHP to send notifications to users of Axis Mobile about new job vacancies that have been posted.


#### **4.4.5 SQLite**

SQLite is a library that implements ‘a small, fast, self-contained, high-reliability, full-featured, SQL database engine’ [15]. SQLite is used in a lot of mobile applications because it is free to use, easy to use and does not need setup or database administration or a server to work [15]. An SQLite database was used to store some data such as the user’s first name, middle name, last name, phone number, email and axis ID which are sent alongside form data for confirmation.

#### **4.5 Axis Mobile**

This section describes how Axis Mobile works with relevant screenshots from the application.

Figure 4.1 shows the first screen that is displayed to a user. It is the log in page and it allows the user to log in with either his Axis ID and password or his email and password. If the “Remember me” checkbox is checked, the application keeps the user logged in so on subsequent use of the application, no login username and password will be required to log in. If the user logs out of the application, then this auto-login functionality does not work.



Axis ID / Email

---

Password

---

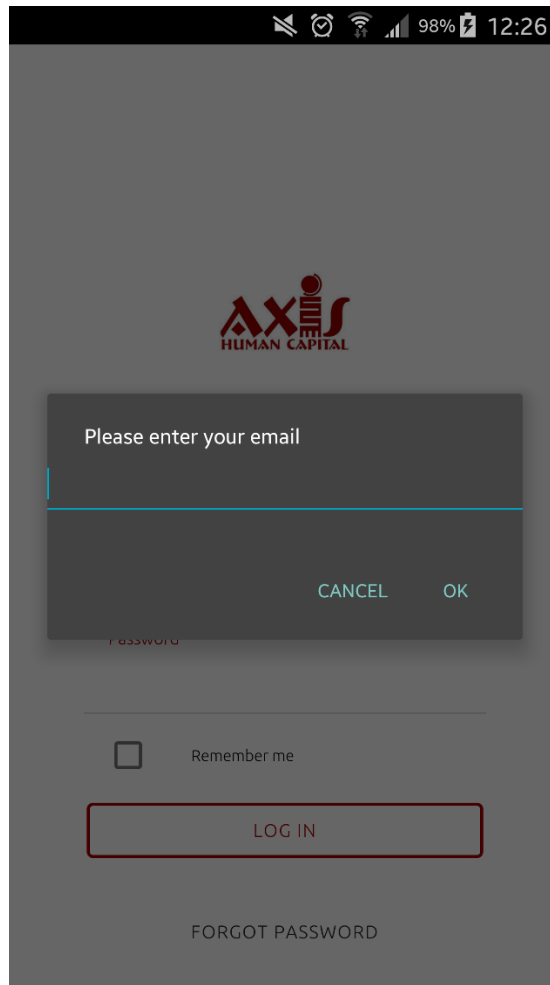
☐ Remember me

LOG IN

FORGOT PASSWORD

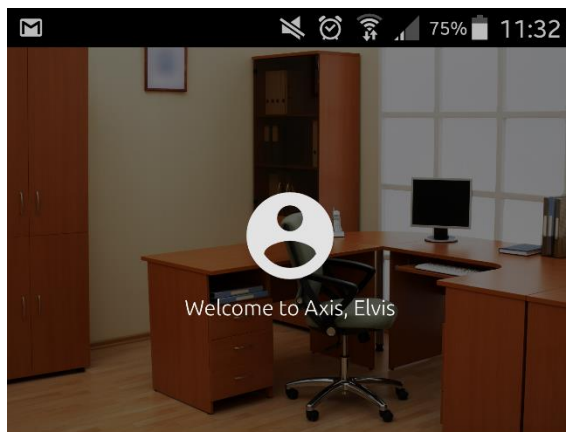
*Figure 4.1: Axis Mobile log in page*

As shown on figure 1.2 below, when a user clicks on the forgot password button, he will be required to enter the email address associated with his account. When he enters the email address, his password is reset, and a new password is generated and sent to his email address. The new password, which is a random set of characters (letters, numbers and punctuations) and hence very difficult to remember, can be reset by the user at any convenient time.

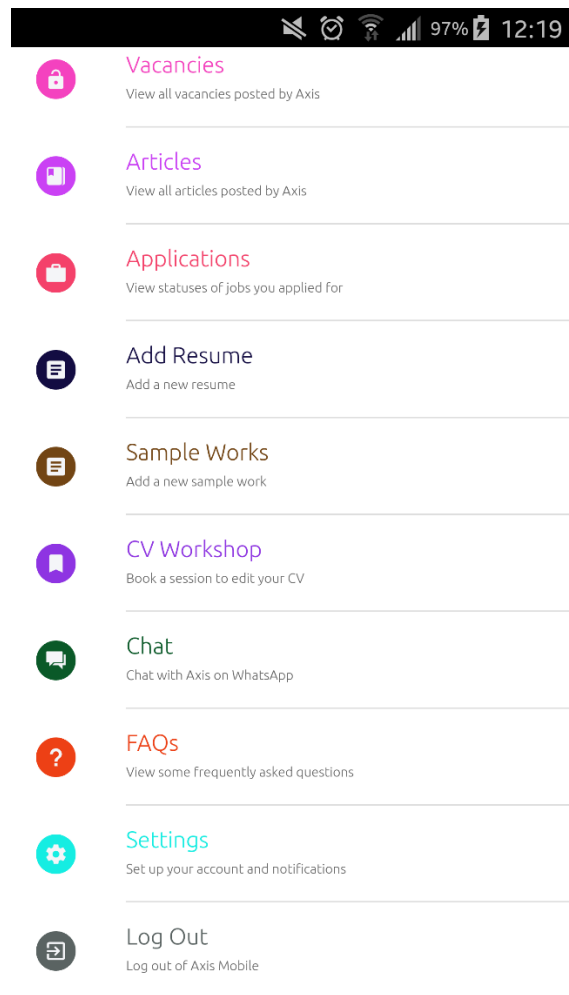


*Figure 4.2 Forgot (or reset) password interface*

When a job seeker successfully logs in, the pages shown in either Figure 4.3a and 4.3b or Figure 4.4 will show. Figure 4.3b is shown when the home page (Figure 4.3a) is scrolled down. The user (job seeker) is welcomed with his first name. The list of actions the user can perform are shown below the welcome picture, each with a brief description to aid first time users of the application. Each list item has a distinct color to help users easily find their most frequently performed action.

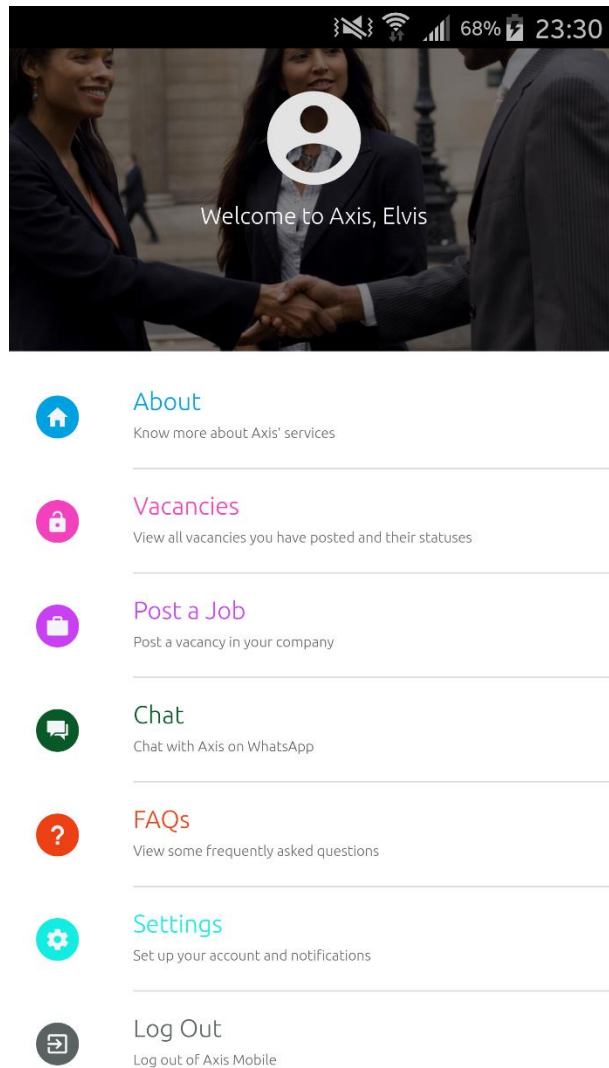


(a)



(b)

Figure 4.3: Home page after a successful login (for a job seeker)



*Figure 4.4: Home page after a successful login (for an employer)*

The job seeker interface allows the job seeker to see quick information about Axis Human Capital Limited and gives him the opportunity to place a call to the company directly from the application. The Vacancies page allows the job seeker to view all jobs that have been posted by Axis and apply to them by uploading a cover letter. The applications page enables the job seeker to see all the jobs he has applied to and the status of each application. The articles page allows the user to read articles that have been posted by Axis on their website. Job seekers are also able to add to their resumes in Axis' database by clicking the Add Resume button. Job seekers who

have written a sample work document can also upload it to Axis' database to increase his chance of getting a job. Job seekers can also book for sessions with Axis to receive help on editing their resumes. With Axis Mobile, job seekers are also able to send direct messages to Axis' WhatsApp number. The FAQs page shows answers to questions that people often ask Axis' employees. The settings page enables job seekers to change their account information such as their names, passwords and secondary emails, and to enable or disable the receiving of notifications on the application. Screenshots from these pages described above are shown in the appendix.

#### **4.6 Choice of Programming Languages for Axis Notification Center**

PHP was the only backend programming language that was considered because it easily integrates into Axis' website which also runs on PHP. PHP was also considered because it is possible to send mails with it, and the notification service provider (OneSignal) also accepts requests from PHP scripts.

HTML, CSS and JavaScript were considered for the frontend of the application because they are supported by all browsers and are very easy to use. The frontend of Axis' website also runs on HTML, CSS and JavaScript.

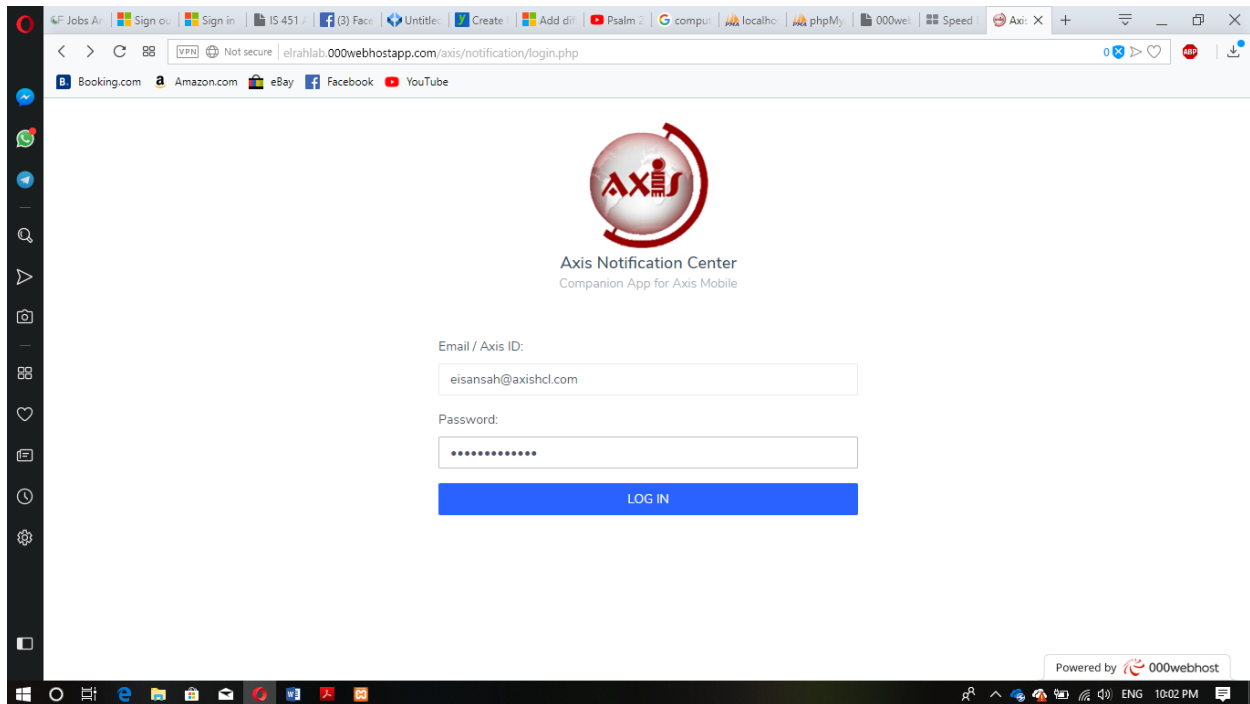
#### **4.7 Choice of Framework for Axis Notification Center**

Another decision that had to be made was whether to use a framework or vanilla code (raw code without the use of a framework). Popular PHP frameworks such as Laravel could make the development much faster, but for a simple application such as Axis Notification Center, vanilla code seemed to work better. Hence, no framework was used in the development of the application.

## 4.8 Axis Notification Center

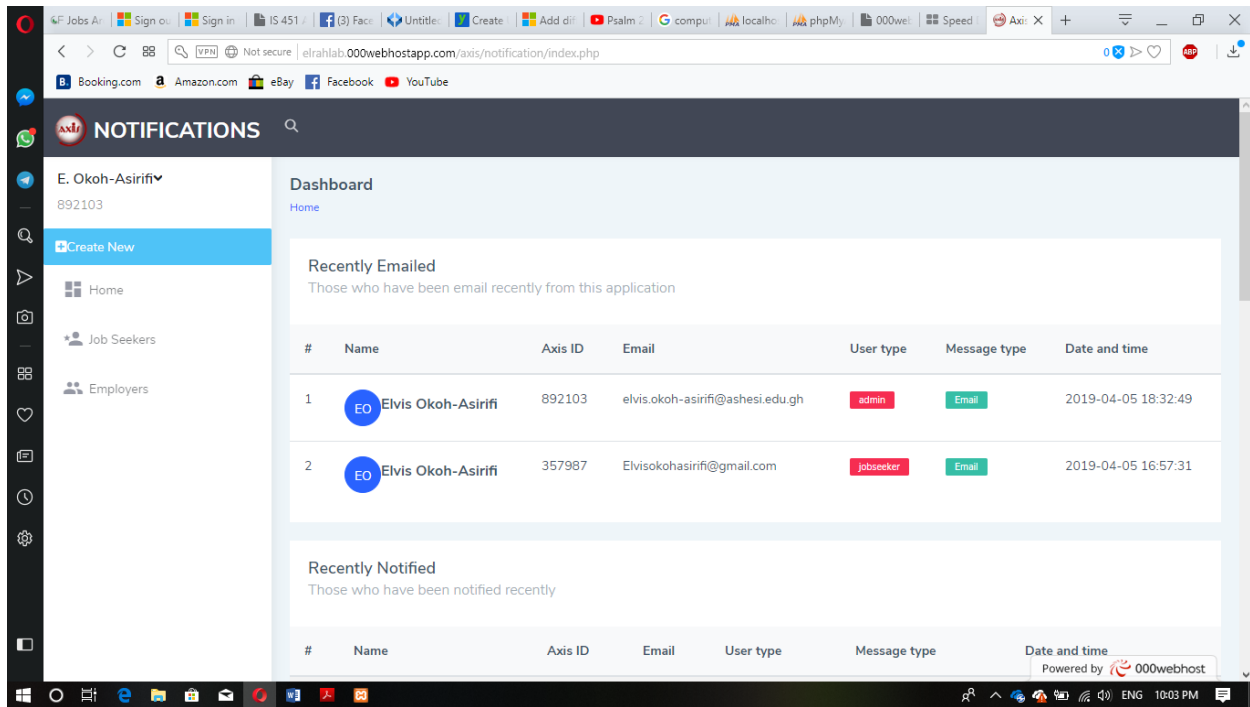
This section describes how Axis Notification Center works with relevant screenshots from the application.

When the user opens the link to the web application, he is presented with the login page where he logs in with his Axis ID or email address and password. This is shown in figure 4.5 below.



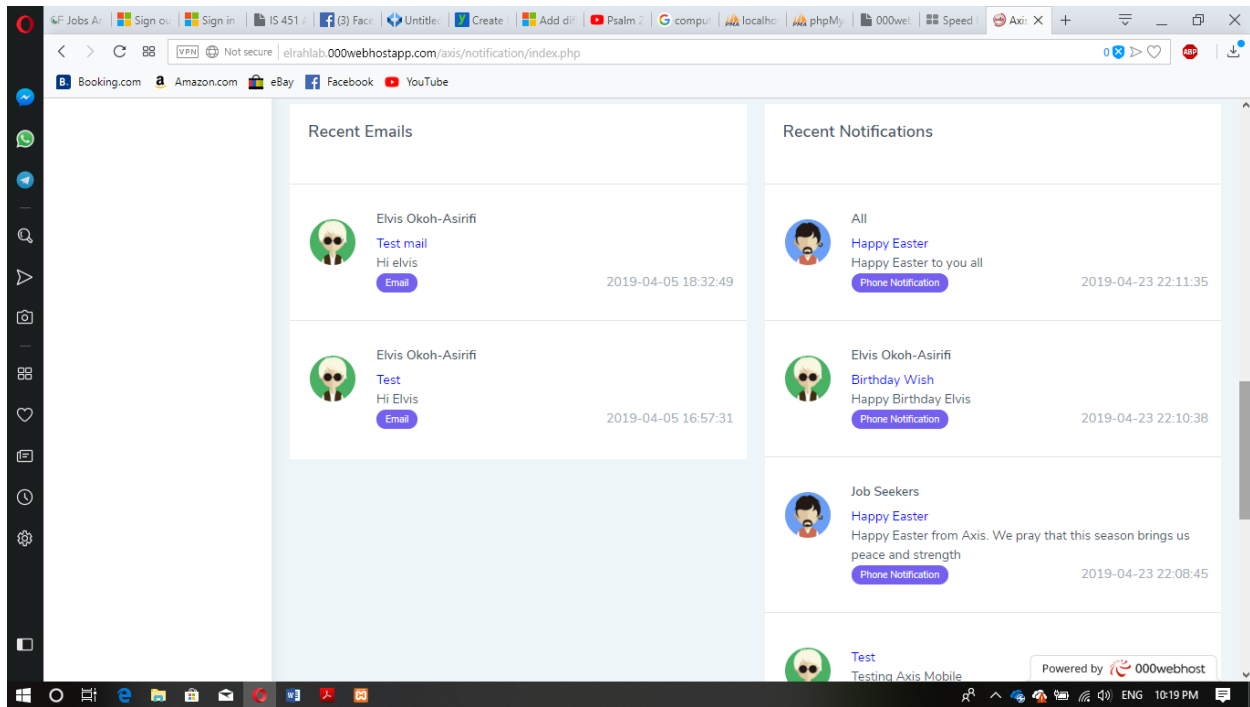
*Figure 4.5: Login Page for Axis Notification Center*

On a successful login, the user sees the main page, which is shown in figure 4.6 below.



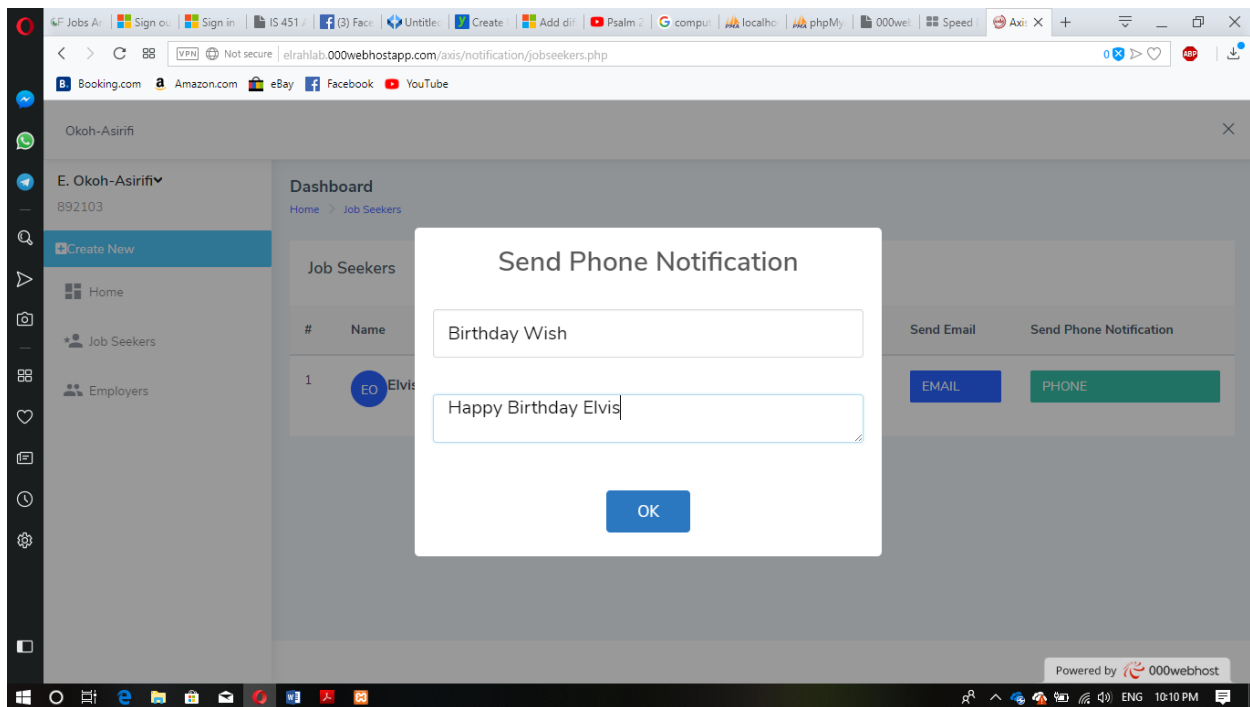
*Figure 4.6: Home Page of Axis Notification Center*

On this page, the user sees some of the previous notifications that have been sent, the type of notification (email or phone notification), which user it was sent to and the date and time the message was sent. When the user scrolls down, he can find the actual messages that were sent. A screenshot to illustrate this is shown in figure 4.7 below:



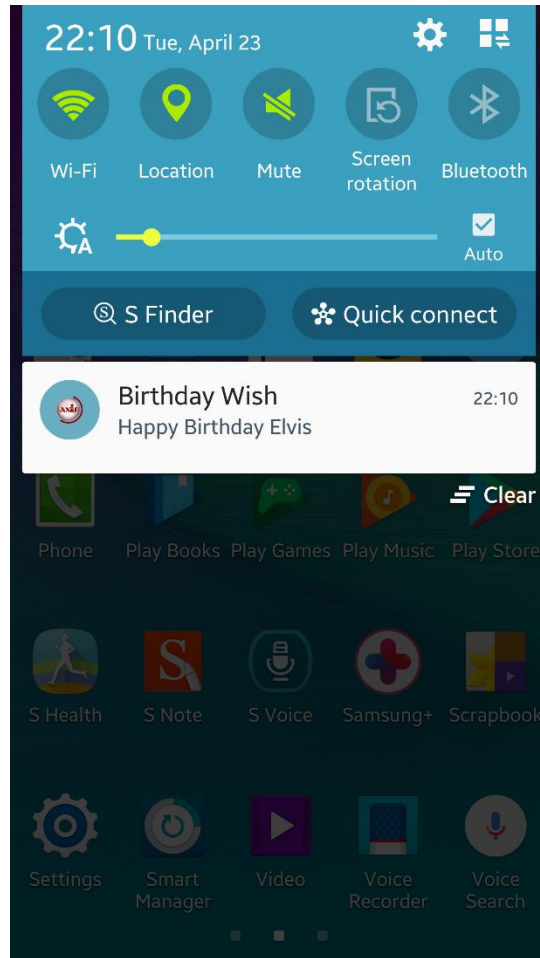
*Figure 4.7: Main Page (scrolled down) of Axis Notification Center*

The user can choose to send a notification or email to a client by searching for the client's name and choosing the type of notification to send (whether email or phone notification). This functionality is shown in the figure 4.8 below.



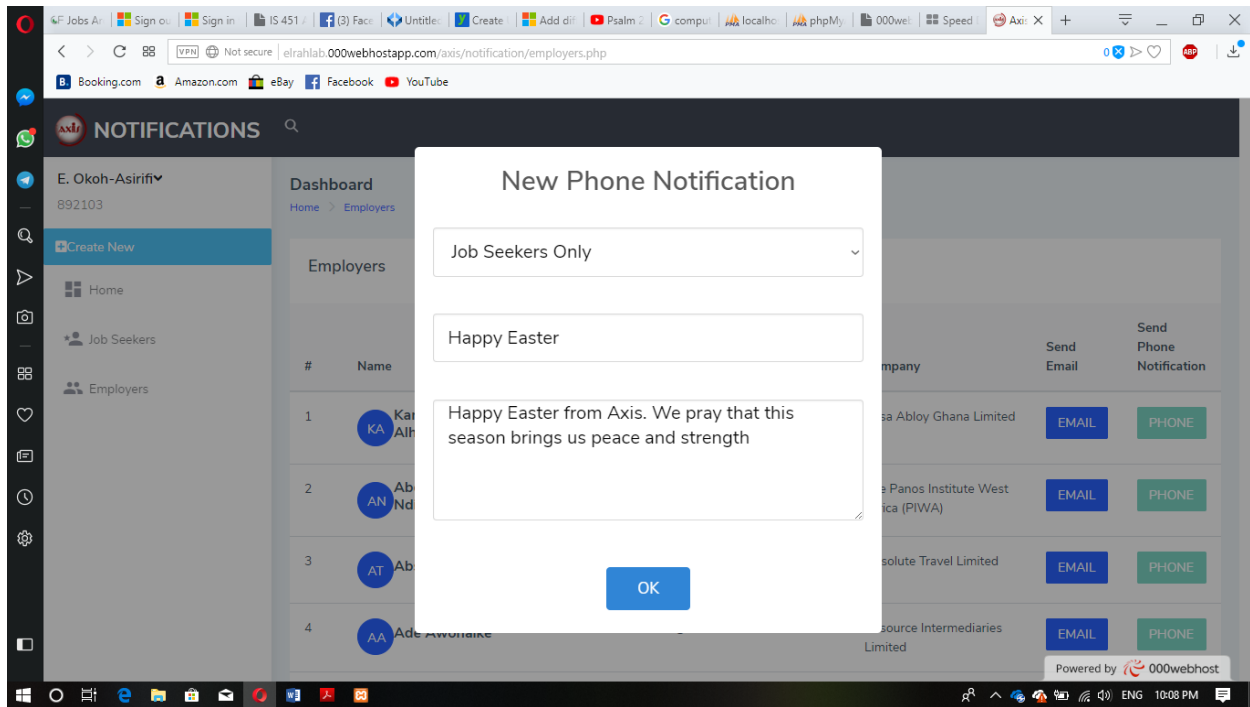
*Figure 4.8: Sending a Phone Notification to Elvis Okoh-Asirifi*

In the screenshot above, the user attempted to send a phone notification to Elvis Okoh-Asirifi, who is a client of Axis. The user first searched for Elvis' name using the search bar at the top of the page. When Elvis' name popped up, the user clicked on Phone under the Send Phone Notification column of the clients table. For clients who have not installed Axis Mobile, the button will be disabled to show that a phone notification cannot be sent to the person. However, an email can be sent to everyone because every client of Axis signed up with their email address. When the user sends the notification, the client receives the notification successfully on his phone (as a pop-up alert if Axis Mobile is already open or as a clickable notification message in the device's notification center if the app is not open). The screenshot of Elvis' phone when he received the phone notification is shown in figure 4.9 below.



*Figure 4.9: Phone Notification from Axis Notification Center*

Axis Notification Center also has a functionality for sending bulk notifications. The user can specify the recipients of the message and send the message according. A demonstration of this functionality is shown in figure 4.10 below:



*Figure 4.10: Sending Bulk Notifications*

In the figure above, the user attempted to send an email to all job seekers. He could have selected that he wanted to send to all employers or all clients.

## **Chapter 5: Testing and Results**

According to Sommerville (2010), the purpose of testing to reveal the presence of errors in the application and ensure that it conforms to users' specifications. Thus, the software is tested to detect situations in which its behavior is undesirable to ensure that it is error-free and that it does what it is expected to do.

### **5.1 Development Testing**

Development testing encompasses all the tests that are made by the developer(s) of the application while they are developing the application to ensure that the application works correctly (Sommerville, 2010). During the development testing of Axis Mobile, unit testing, component testing and system testing were conducted.

#### **5.1.1 Unit testing**

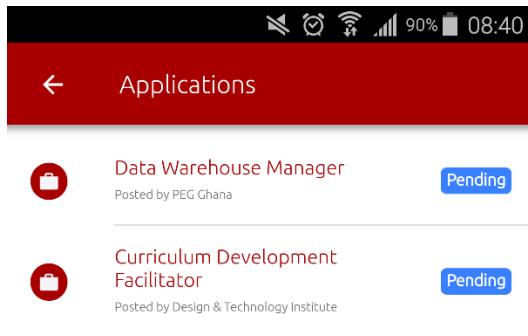
Unit testing involves testing the individual functions or program units to ensure that they work correctly (Sommerville, 2010). During development, after every function was created especially on the server side, the function was tested with possible user inputs (both valid and invalid inputs) to ensure that it worked correctly given the inputs. Functions such as the function for selecting job openings, logging in, password reset, sending emails, selecting jobs applied to by a particular job seeker, sending notifications and selecting jobs posted by an employer were all tested individually on the server side separately to ensure that they worked very well.

Functions for validating user inputs on the client side were also tested on various data inputs to ensure that the application worked well in all situations. For instance, on the vacancies page where the job seeker submits his cover letter to apply for a job, the cover letter format should either be a Microsoft Word file or a PDF file. All other file types are unacceptable. These

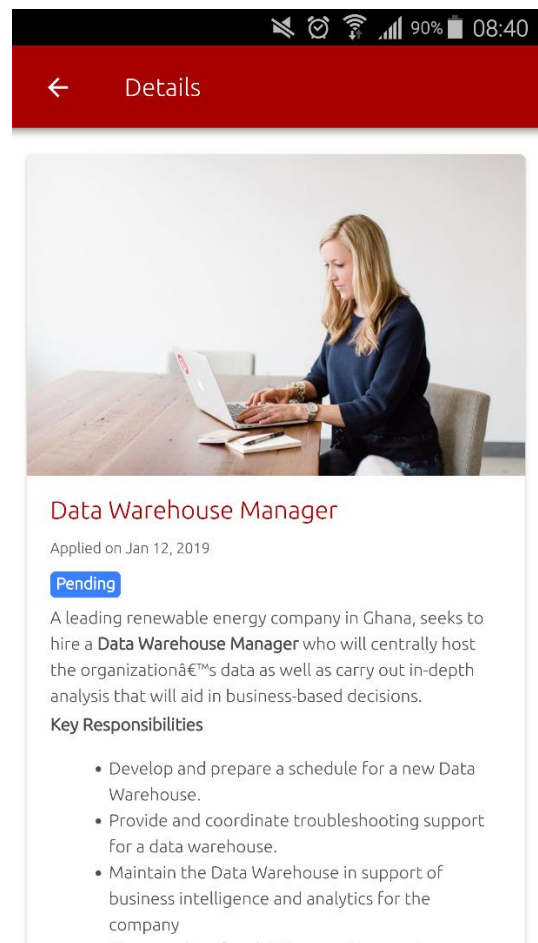
validation functions were tested on different smartphones to ensure that they work on all platforms and devices.

### **5.1.2 Component testing**

Component testing involves testing individual units that have been ‘integrated to create a composite component’ [6:210]. During the development of Axis Mobile, component testing was done by testing all interfaces that pass data to other interfaces. For instance, on the applications page where users can view the status of the jobs they have applied to, clicking on any of the jobs opens another page that enables the user to view details of that job. These actions are demonstrated in the figures below (Figure 5.5 and Figure 5.6):



(a)



(b)

Figure 5.1: Job Status (a) and Job Details (b) pages

From figures 5.1a and 5.1b, when the Data Warehouse Manager job was clicked, it opened a details page that gave an overview of the job that had been applied for.

All pages that pass data to other pages were tested to ensure that they passed the correct data every time.

Component testing was conducted on Axis Notification Center to ensure that all pages were linked correctly and that the data passed from each page to the next was correct.

Moreover, all functions that pass data to other functions within the application were also tested to ensure that the passed data was correct and worked well in the receiving functions as well.

### **5.1.3 System Testing**

System testing involves putting all the components together and testing all of them as a whole.

Sommerville [6:219] explains that system testing ensures that all the components are ‘compatible, interact correctly and transfer the right data at the right time across their interfaces’.

During system testing of Axis Mobile, the complete application was built and used on a Samsung Galaxy Note 4 phone to test how the entire application works. All bugs that were discovered were quickly fixed and the application was built again and tested to discover other bugs. This iterative process continued until the developer thought the first version of the application was ready for user testing.

Axis Notification Center was also tested on Opera, Mozilla Firefox, Google, Safari and Microsoft Edge (which are popular browsers) to ensure that it works appropriately on all the browsers.

## **5.2 Release Testing and User Testing**

Release testing means testing a released version of an application by people who are not on the development team [6]. The goal of release testing is to ensure that the application is good enough for use.

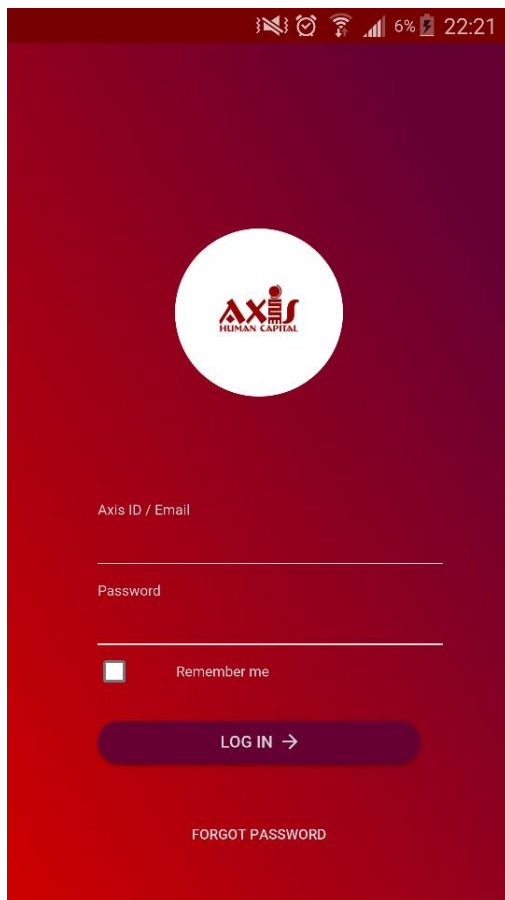
User testing is the stage where users of the application use the application and provide inputs or feedback on the application [6].

Beta testing was conducted on the two applications after development testing was done (as release and user testing).

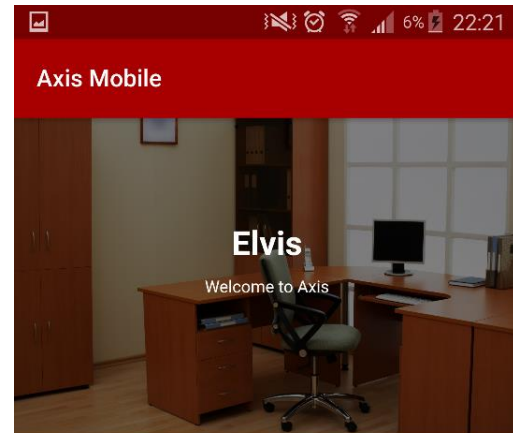
### 5.2.1 Beta testing

Beta testing involves making a release of the application available to users to use and raise problems they encounter while using the application [6].

After the first version of Axis Mobile was developed (developed using Ionic version 3), employees at Axis were asked to download, install, test and give feedback on the functionality and user experience of the application. The application's login page and home page looked like the figures below (Figure 5.2 a and b):



(a)



(b)

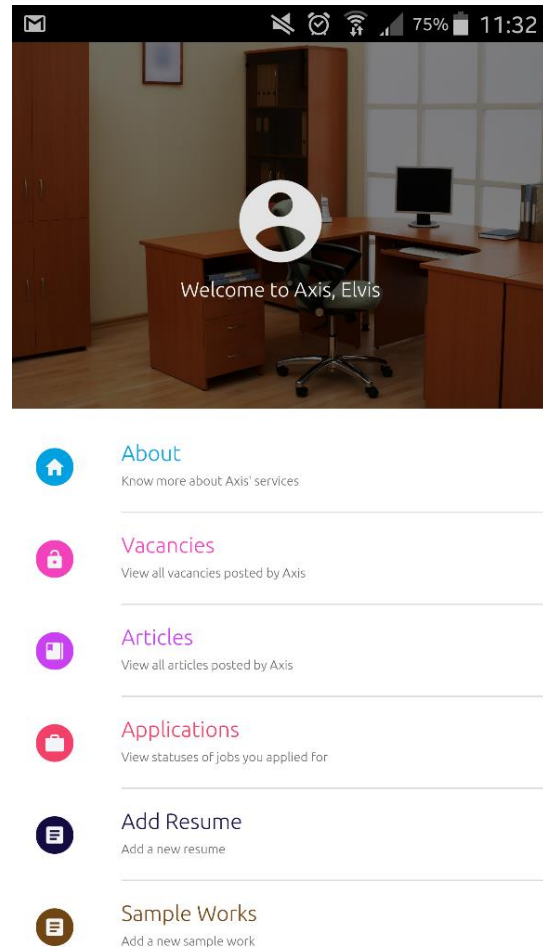
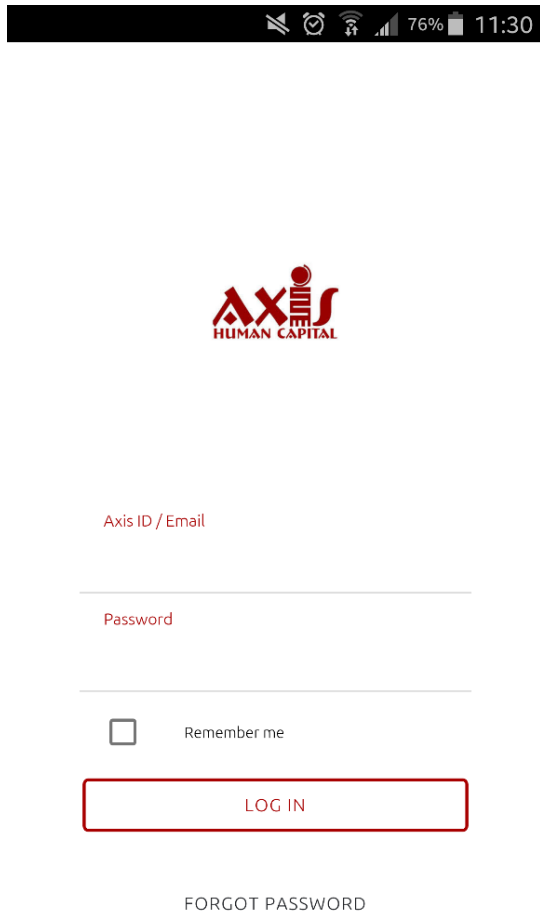
*Figure 5.2: Login and Home pages of the first version of Axis Mobile*

One feedback that was received was that the application was ‘too red’. Thus, the red color used on the log in page and other pages was too much and that it made the application look ‘dangerous’ because red connotes danger in the Ghanaian culture. The application also seemed to be challenging to use for first-time users because all those who tested the first version of the application requested for a walkthrough document. Also, those who tested the application complained about the startup time of the application. A few of them mentioned that it could take up to ten seconds before they would see the login page of the application (the first page of the application).

To mitigate these problems, a second version of the application was created using Ionic version 4. The Ionic version 4 has a faster startup time than the version 3, and it was able to reduce the startup time from about ten seconds to about two to three seconds, which was a significant improvement.

The colors were also changed on the pages. Most of the ‘reds’ were replaced with white to reduce the redness of the application.

To further ensure that first-time users would not require a walkthrough to be able to use the application to the fullest, the buttons on the home page were changed from a grid view to a list view with an explanation for what each list view item (or button) does. The improvement is shown in the image below (Figure 5.3):



(b)

Figure 5.3: Login (a) and Home (b) pages

The second version of the application was also released for testing. Users of the application mentioned that the new look of the application and the startup time were a lot better than the previous one and that the brief description of the list view items made them quickly know what each button did. However, they mentioned that the feedback they received when they performed certain actions was not detailed enough. For instance, when they tried to reset their passwords and the reset was successful, they would see an alert that would inform them that their passwords have been reset successfully but it would not inform them of their new password or what it had

been changed to. Also, on pages such as the vacancy page where job seekers can apply for jobs by submitting their cover letters, users of the application suggested that the label on the button could be changed from 'SAVE' to 'SUBMIT'. They suggested this because a 'SAVE' button seemed to mean that the cover letter was just going to be saved on the mobile device without being submitted.

The feedback provided by the initial users of the application were considered and the issues they raised were all addressed before the final version of the application was released for use.

Beta testing was also conducted on Axis Notification Center, which was built after Axis Mobile had been released for beta testing. All the issues raised during the testing of Axis Mobile were considered when creating Axis Notification Center. The application has been tested by employees of Axis and no errors have been reported yet.

## **Chapter 6: Conclusion**

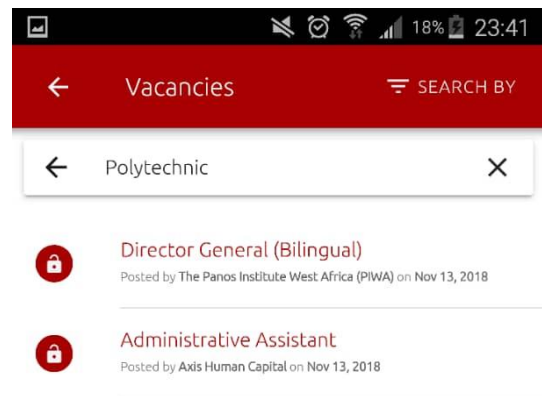
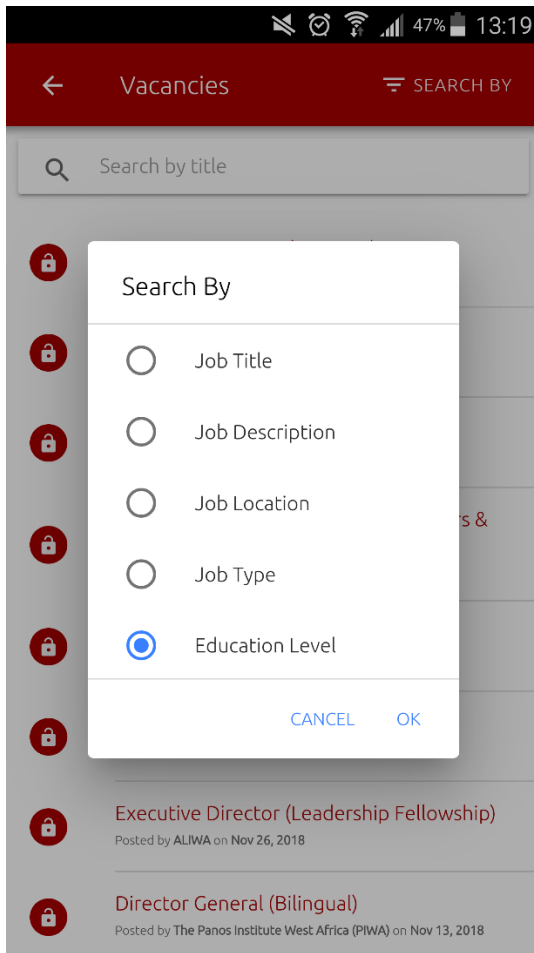
This chapter explains how Axis Mobile and Axis Notification Center support Axis' processes and how they enhance users experience with Axis systems. Recommendations for future versions of the application are also explained in this chapter.

### **6.1 How Axis Mobile will support Axis' Processes**

- It will reduce the number of calls and emails Axis receives from job seekers by a significant percentage. Employees of Axis explained that they receive a lot of emails from job seekers who have forgotten their Axis IDs or their passwords, which are needed for accessing the job seeker dashboard. However, Axis Mobile enables users to log into the application using either their email addresses (which Axis believes are easier to remember) or their Axis IDs. Axis Mobile also has a feature that allows users to remained signed in to the application without the need to enter their log in details again until they sign out. The password reset option in Axis Mobile, which takes less than a minute to reset a user's password, will also enable Axis' users to reset their passwords much more easily without the need to send Axis a mail or the need to place a phone call to Axis.
- Axis can now send quick messages to all their clients and monitor how many of them read the message. Axis Mobile uses OneSignal APIs to send notifications to all clients who have downloaded and installed the application. OneSignal will allow Axis to send important messages to clients' devices at no cost at all within second and these messages will be received and read using Axis Mobile. The number of people who have read the message can also be tracked by Axis on their OneSignal dashboard.

## **6.2 How Axis Mobile will enhance users' experience with Axis Human Capital**

- With users being able to easily reset their passwords using the mobile application, there will be no need to place phone calls to Axis or send an email to Axis to get passwords reset. The passwords can be reset anytime and anywhere.
- Users can save some data for other useful activities. Users can opt for a one-time sign option on Axis Mobile, ignoring the need for page loads on the website which would also consume some data. An average user would open the Axis website, open the login page, log in and then is redirected to the dashboard. It is after he has logged in that the user can view and apply for jobs. Axis Mobile removes the need for loading the Axis home page, the login page (if the user opts for single sign in) and the dashboard page (Axis Mobile does not use data to load the dashboard page).
- Job seekers can easily search for available jobs that suit their needs and apply to them using Axis Mobile. For example, on Axis Mobile, a job seeker can find all full-time jobs or all job openings within a particular city or all jobs that meet his highest education. This feature, which is only available on Axis Mobile and not on Axis' website, is shown in the figures below and it enhances the way users find jobs that match their qualifications and interests.



*Figure 6.1: Search functionality on Axis Mobile*

- Finally, clients who wish to book sessions with Axis are only required to provide the date, time and specify their employment status when booking sessions with Axis Mobile because their names and phone numbers are cached on their devices. On the website, they would have to provide their full names and phone numbers as well. This quickens the rate at which users can get things done.

### **Recommendations for future work**

Axis Mobile can be made better if:

- users are able to chat with employees of Axis using multiple chatting platforms such as Skype, Telegram, Viber, etc. rather than limiting them to WhatsApp.
- a chatbot is embedded in the application to help respond to some of the messages Axis' employees will have to respond to, considering their large customer base. This will ensure prompt response to users' queries.
- native applications are built to replace the current hybrid application. Thus, Axis can develop a native Android application for their clients who use Android devices, a native IOS application for the users that use IOS devices and a native Windows application for their clients who use Windows devices. This is because native applications have a faster startup time and generally perform better than hybrid applications.

## References

1. Dave Bartram. 2000. Internet Recruitment and Selection: Kissing Frogs to find Princes. *International Journal of Selection and Assessment* 8, 4: 261–274.
2. Charles R. Greer, Stuart A. Youngblood, and David A. Gray. 1999. Human Resource Management Outsourcing: The Make or Buy Decision. *The Academy of Management Executive (1993-2005)* 13, 3: 85–96.
3. Anna B. Holm. 2012. E-recruitment: Towards an Ubiquitous Recruitment Process and Candidate Relationship Management. *Zeitschrift für Personalforschung / German Journal of Research in Human Resource Management* 26, 3: 241–259.
4. Ionic. Ionic - Cross-Platform Mobile App Development. *Ionic Framework*. Retrieved April 1, 2019 from <https://ionicframework.com/>.
5. Hong Kesen. 2012. How e-HR optimizes the human resource management process. *Engineering Design and Manufacturing Informatization 2012 3rd International Conference on System Science*, 369–372.
6. Ian Sommerville. 2010. *Software Engineering*. Addison-Wesley, New York.
7. Axis Human Capital Limited. Retrieved April 1, 2019 from <https://axishcl.com/home3/>.
8. Bamboo Testimonial. Retrieved October 10, 2018 from <http://www.bamboohr.com>.
9. Workforce Ready Suite. *Kronos Inc.* Retrieved October 10, 2018 from <https://www.kronos.com/products/workforce-ready-suite>.
10. Recruitment Software | Applicant Tracking System - Zoho Recruit. *Zoho*. Retrieved October 10, 2018 from <https://www.zoho.com/recruit/index.html>.
11. ICIMS Talent Platform - Talent Acquisition Software. Retrieved October 10, 2018 from <https://www.icims.com/talent-acquisition-software>.
12. PHP: Hypertext Preprocessor. Retrieved April 1, 2019 from <https://www.php.net/>.
13. phpMyAdmin. *phpMyAdmin*. Retrieved April 1, 2019 from <https://www.phpmyadmin.net/>.
14. Product Overview. Retrieved April 1, 2019 from <https://documentation.onesignal.com/docs>.
15. SQLite Home Page. Retrieved April 1, 2019 from <https://www.sqlite.org/index.html>.

## **Appendix**

### **A.1 Interview Questions**

#### **A.1.1 Employees of Axis:**

- Considering the current system you are using (website), what do you think is a challenge to both job seekers and employers?
- What functionality do you think will help solve this challenge?
- What other features will help job seekers and employers?

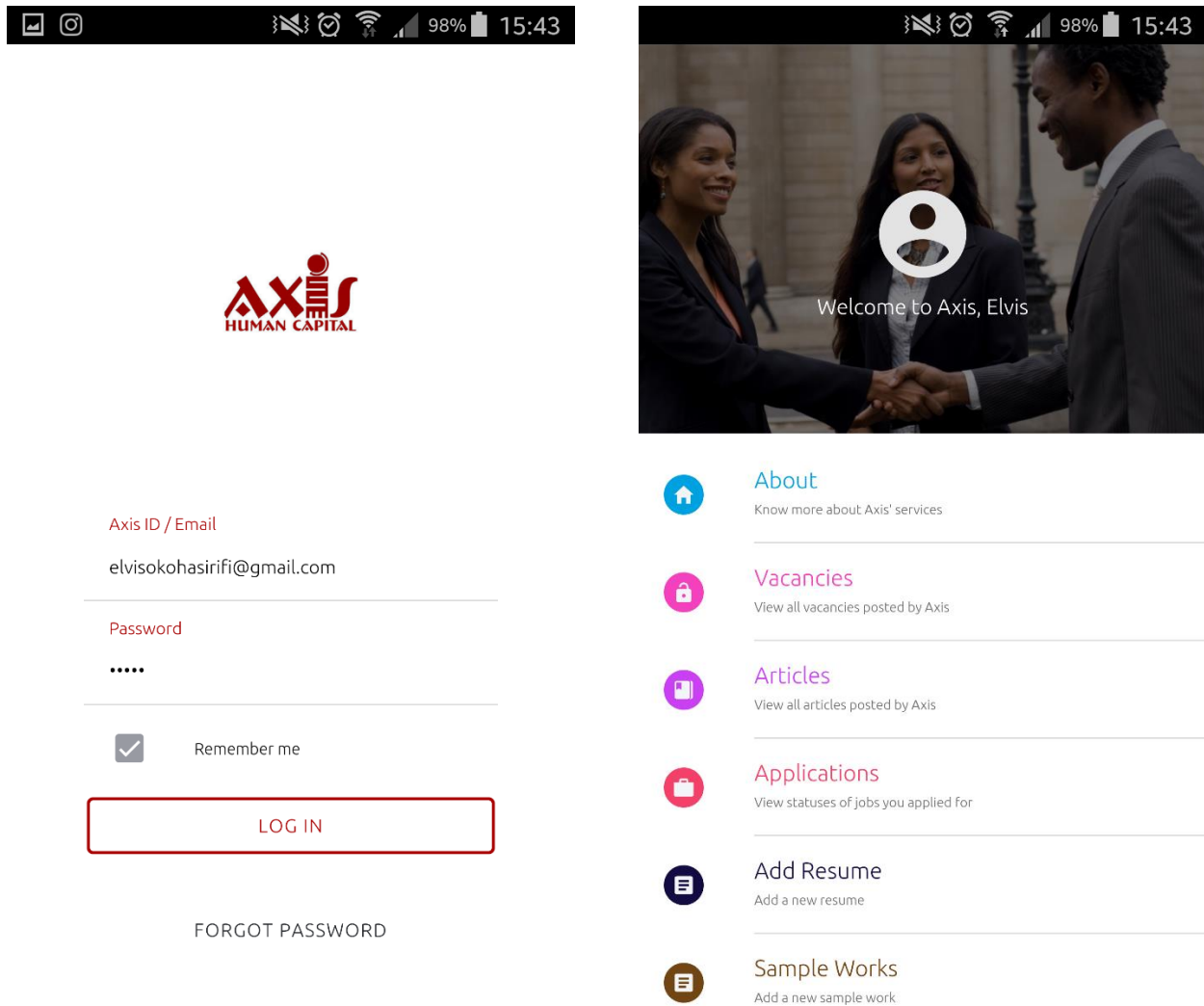
#### **A.1.2 Job seekers:**

- What information do you often look for on the Axis website?
- Does the website work very well for you? Does it fulfill all your needs?
- Which would you prefer: submitting your resume to be edited in person or submitting it online and getting it edited at the same fee?
- Would you like to have a chat feature in the mobile application or you want the chat feature to be synchronized with a chat application already installed on your phone?
- What kind of notifications will you like to receive? (notifications on new job opportunities, training podcasts and videos, other general announcements etc.)
- What other feature(s) would you need in the application?

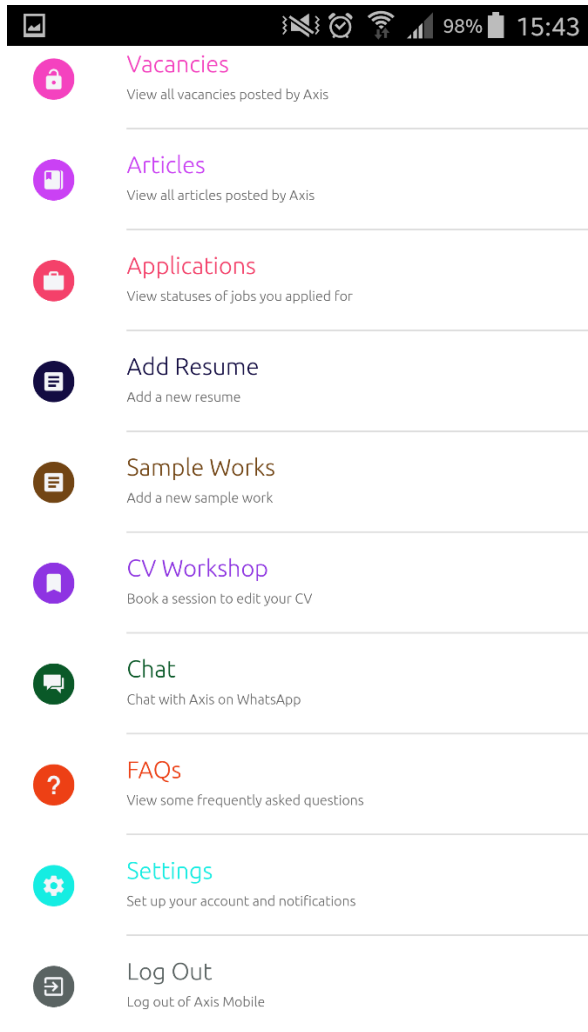
#### **A.1.3 Employers:**

- What operation do you often perform with the already existing system (website)?
- Does it fulfill all your needs? Which ones are unmet?
- What feature(s) would you need in a mobile application for Axis?

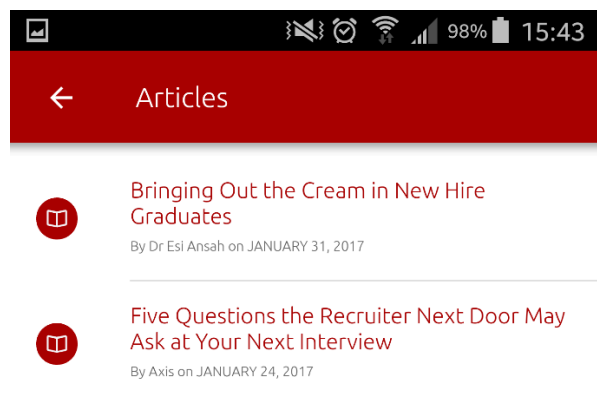
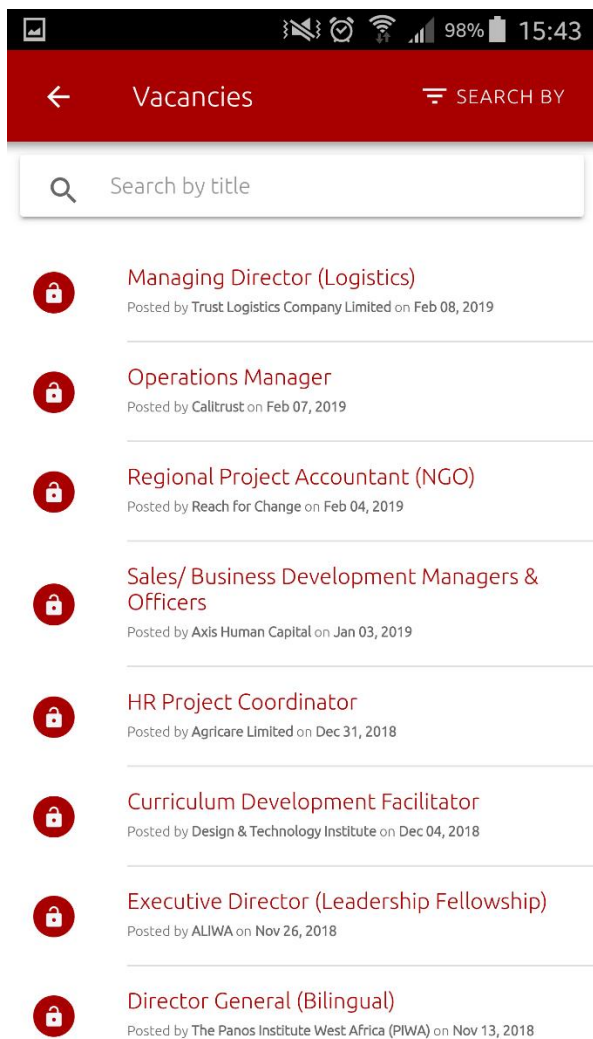
## A.2 Screenshots of Axis Mobile



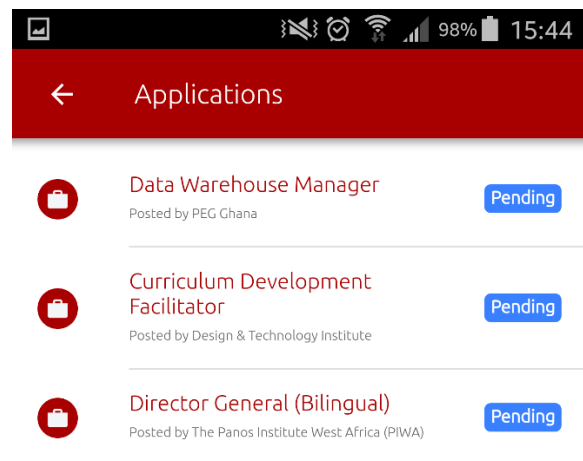
*Figure A.1 Login page (left) and Home (right) page*



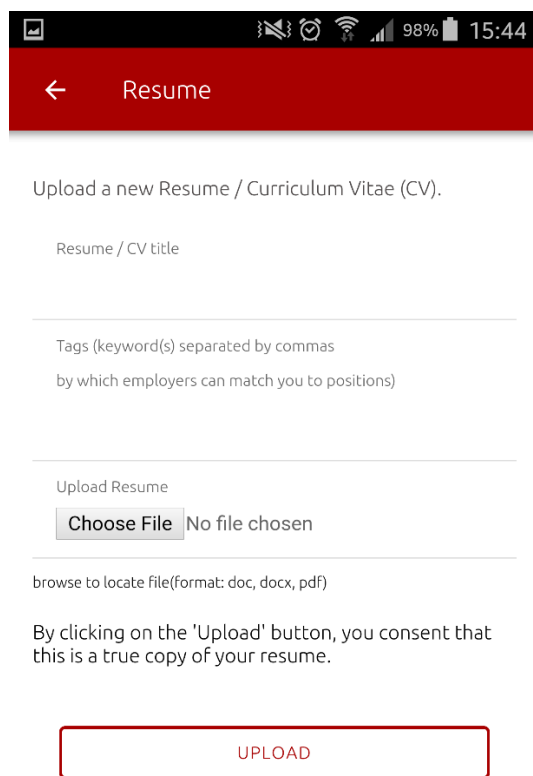
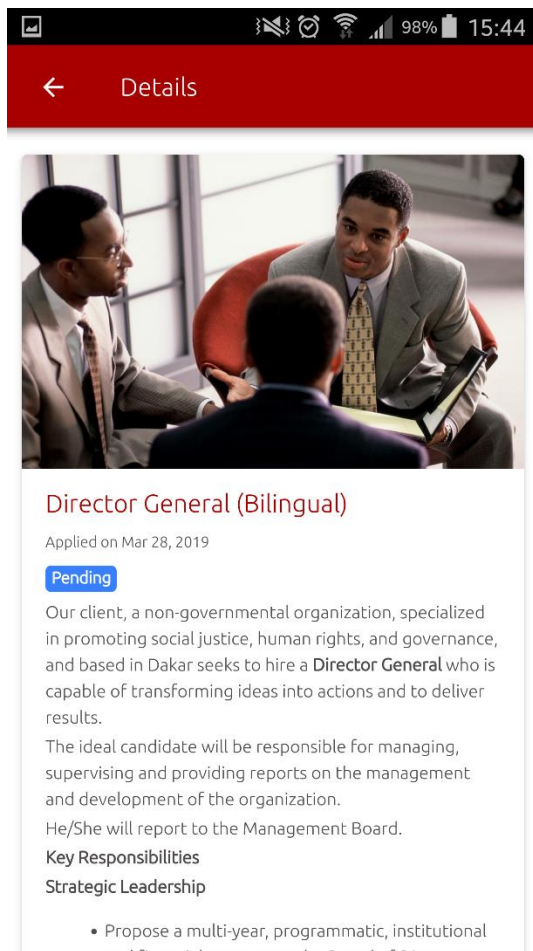
*Figure A.2 Home page scrolled down (left) and About page (right)*



*Figure A.3 Vacancies page (left) and Articles page (right)*



*Figure A.1 Reading Lounge (left) and Job Applications page (right)*



*Figure A.5 Job Application Details page (left) and Apply for Job Page (right)*

Sample Work

Upload a new sample work.

Sample work title

---

Upload Sample Work

**Choose File** No file chosen

---

browse to locate file(format: doc, docx, pdf, xls, xlsx, ppt, pptx)

By clicking on the 'Upload' button, you consent that this is a true copy of your sample work.

**UPLOAD**

CV Workshop

jobseeker, or need it for some other specific purpose. The first session is at 11:00am prompt, and the second is at 2:00pm prompt. Each session lasts one and a half hours. There are only FIVE slots per session, and so do sign up now! Fill the form below, and check your email for full details.

Preferred Date

---

Time Available

11:00 AM

---

Are you currently job hunting?

Yes

---

Current Work Status

Full Time

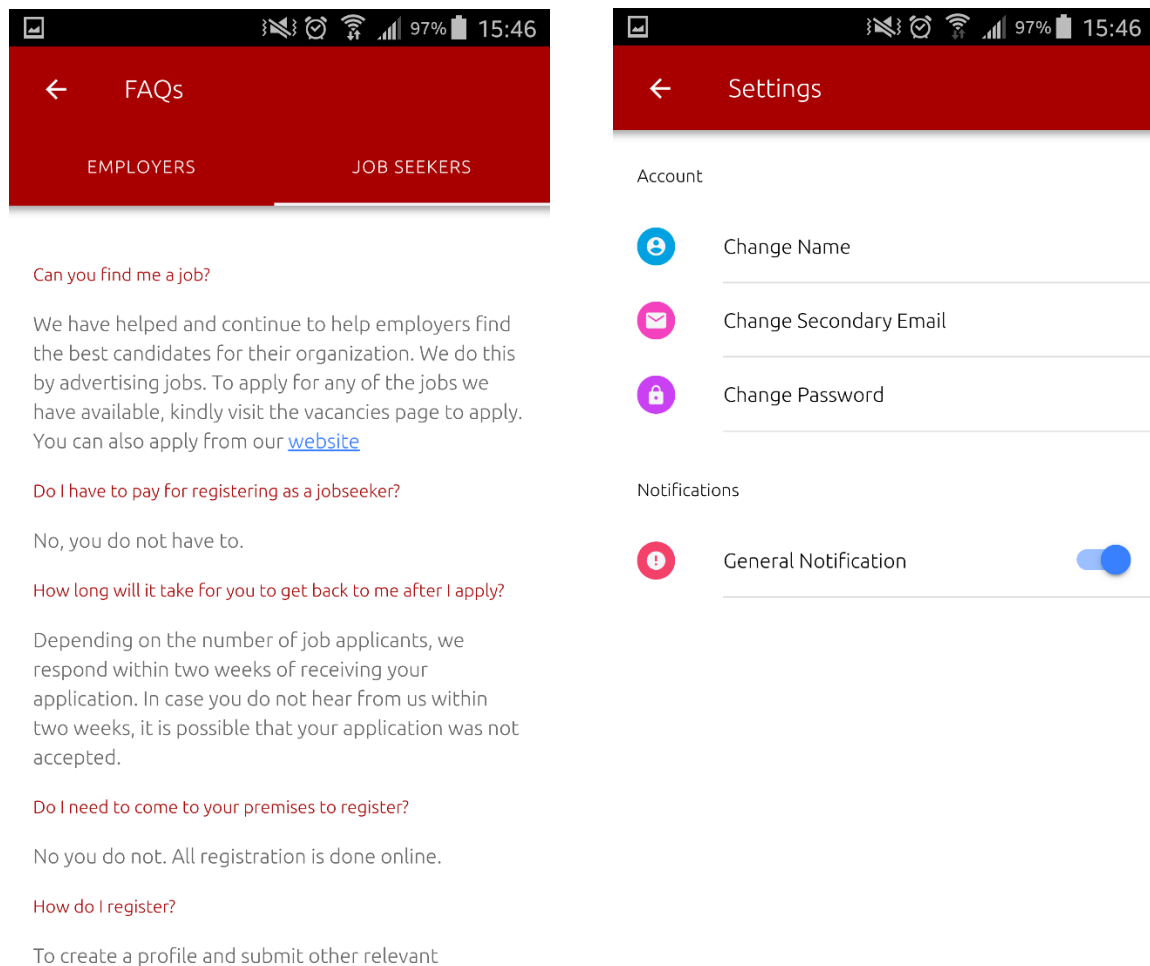
---

Additional Information

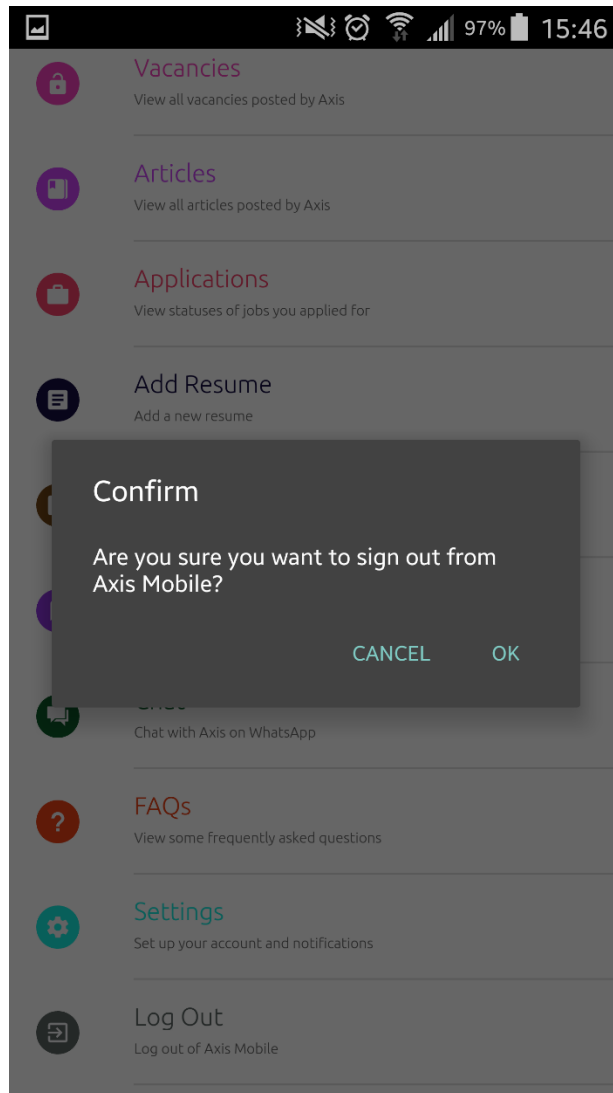
---

**SUBMIT**

*Figure A.6 Sample Work Page (left) and CV Workshop Page (right)*



*Figure A.7 FAQs (left) and Settings Page (right)*



*Figure A.8 Logout screen*