



Development of a Software System for Realtime Management of Crime Reports in Southwestern Nigeria: The Administrative Approach

Joseph Akinyede^{1,3}, Akinlolu Ponnle², Chinedu Olebu^{2,*}, Francis Akinluyi³, Aderonke Thompson⁴, Olurotimi Dahunsi⁵, Boniface Alese⁴, Michael Oyinloye⁶

¹Centre for Space Research and Applications, Federal University of Technology, Akure, Nigeria

²Department of Electrical and Electronics Engineering, Federal University of Technology, Akure, Nigeria

³Department of Remote Sensing and Geoscience Information System, Federal University of Technology, Akure, Nigeria

⁴Department of Cybersecurity, Federal University of Technology, Akure, Nigeria

⁵Department of Mechatronics Engineering, Federal University of Technology, Akure, Nigeria

⁶Department of Urban and Regional Planning, Federal University of Technology, Akure, Nigeria

Email address:

cgolebu@futa.edu.ng (Chinedu Olebu)

*Corresponding author

To cite this article:

Joseph Akinyede, Akinlolu Ponnle, Chinedu Olebu, Francis Akinluyi, Aderonke Thompson, Olurotimi Dahunsi, Boniface Alese, Michael Oyinloye. Development of a Software System for Realtime Management of Crime Reports in Southwestern Nigeria: The Administrative Approach. *American Journal of Science, Engineering and Technology*. Vol. 8, No. 1, 2023, pp. 23-32. doi: 10.11648/j.ajset.20230801.13

Received: December 13, 2022; **Accepted:** January 25, 2023; **Published:** February 16, 2023

Abstract: One of the major challenges of managing crime data is lack of robust and adequate management system. In this work, a comprehensive administrative website was built for effective management of crime reports/data. The crime data which was collected from a dedicated reporting mobile application was rendered into different categories on the admin dashboard. The admin dashboard incorporates features such as multilevel accessibility (which includes the generic admin and super-admin), an option to create reports, view reports, add a specific security outfit and so on. Unified Modeling Language (UML) and the waterfall software development technique were used in building the administrative website. The client side of the website was developed with client-side technologies, namely HTML, CSS, Vanilla JavaScript, and React, while the server side of the app was built using Node JS and MongoDB database. The application project files were hosted on MVPS virtual private cloud server, which is characterized by virtualization of resources and provision of backups. The developed software system was evaluated using the Lighthouse chrome developer tool and beta testing. The results of the Lighthouse analysis showed that the website is responsive and easily accessible. On the other hand, the results obtained from the beta test showed that the users' experience was not hampered while the web application was being put to use.

Keywords: UML Class Diagram, UML Sequence Diagram, Waterfall, User Dashboard, Administrative Pages, Performance Evaluation

1. Introduction

The introduction of microblogging platforms and messaging applications (such as WhatsApp, Instagram and Facebook) into the technology space has made sharing information (in the form of video, audio and texts) seamless. Human activities can now be recorded in real-time and shared between closed groups on the fly. Superficially, this

would mean that the rate of criminality would reduce since already-established national security architectures [including the Nigeria Police Force (NPF) and National Security and Civil Defence Corps (NSCDC)] would now have first-hand knowledge about crime activities within a particular locality or community. In reality, this is not the case due to

ineffectual management systems and architectures and lack of the required synergies between different security outfits. In recent times, national security architectures have achieved giant strides in securing lives and properties. Unfortunately, coordinated crimes such as terrorism and banditry appear to be out of the radar of these security architectures. Although the Nigerian media have attributed these security challenges to some political elements within the country, establishment of an all-encompassing management system for crime reports and activities will suffice and assuage the prevailing challenges.

Information technology (IT) has gained wide acceptability in the management of crime in most advanced countries of the world. Despite the fact that IT has become notoriously popular amongst Nigerians, especially the young populace, it has not been quite accepted in the area of crime management and monitoring. In fact, a preliminary survey conducted in Calabar metropolis, where descriptive analysis of sample responses provided by 200 respondents were analysed, showed that IT is yet to be embraced as a method of crime management in Nigeria [1]. As a result, this study proposes a comprehensive management platform entitled *Geotiscm* which effectively manages crime and provides the required synergies between security architectures. This study spotlights the software development cycle, and the technologies that were utilized throughout the development phases of the software system. The rest of this study is structured in the following format: Section II (Literature Survey), which explores the state-of-the-art technological solutions for crime management and presents the proposed solution, Section III (Material and Methods), shows how the proposed crime management system was developed from the ideation stage to deployment. This section is followed by the Results and Discussion, where we carry out performance evaluation of the system. This study culminates in Conclusion and Recommendation, which summarizes the entire work, and suggests ways of improving the solution.

2. Literature Review

Researchers, individuals, and stakeholders have made several attempts to develop crime management systems that will be well suited to the current reality of the security situation in the country. For instance, “an intelligent crime management system for Lafia metropolis” was developed by Jeffrey O. A. *et al.* [2]. In their work, legacy crime data was factored in which was then enriched with data obtained from crime reports. While it was reported that the proposed system performed well in predicting the likelihood of a crime taking place based on historical data, the system did not support multi-level security architectures and is only confined to one location (Lafia). Similarly, an online crime reporting system was proposed and directed only at community policing, which is one of the shortcomings of the proposed system [3]. Another crime management system was developed using

PHP-based logic and its performance was measured using ISO/IEC 25010:20011 [4]. The developed system, however, is user-centric and does not really incorporate crime management. In the same vein, an online crime management system, which used the PHP language was developed using Pakistan as a case study [5]. The use of PHP in developing the management systems as proven by Tomas U. G. *et al.* [4] and Saqib N. *et al.* [5] is a major weakness because PHP has some security vulnerabilities, such as SQL injection, Cross-site Request Forgery (CSRF), PHP Object Injection, Session Hijacking and Stream Injection Attacks [6]. A web-based only crime data management system was also developed by Manisha G. I. *et al.* [7]. Although the system serves as a means of reporting crime and managing crime data, its major shortcomings stem from the issue of separation of concerns and limited coverage (because it targets only the police security architecture and excludes other security outfits). Another solution was proposed by Oludele A. *et al.* [8], in which a computerized real-time crime management system (CRMS) was developed for managing crime data. This solution was intended to aid security agencies in managing crimes and carrying out background checks of criminals [8]. The authors anticipated that national security will be improved with this solution. Notwithstanding the advantages of this system, even though other similar crime databases, such as Nigeria Data Portal exist, the security of Nigeria has worsened. This implies that having a compendium of crime data will not provide a lasting solution to the prevailing security situation of Nigeria.

In this work, an effective and transparent crime management system was introduced. The system was inspired by following the approach of the British Council for “introducing a crime management system” [9]. The proposed system emphasizes proactivity, clarity, and transparency by adopting a victim-centered approach, which could potentially improve citizens’ confidence in Nigeria’s security architecture and policing services.

3. Methodology

In this study, the proposed system was implemented using the Waterfall software development model. The Waterfall model (Figure 1) is a sequential software development technique that progresses steadily from project requirements analysis to maintenance of the deployed application [10]. The model was adopted to simplify the documentation of the software development project in advance because of its clear structure, and ease of information transfer [11].

3.1. Requirement Analysis

In this section, the requirements of the proposed system (administrative website) were identified (Figure 2). These requirements represent a low-level overview of the proposed administrative website. The system architecture of the proposed administrative website is shown in Figure 3. Its major components are web-based crime management systems, security agents and an observer/eyewitness.

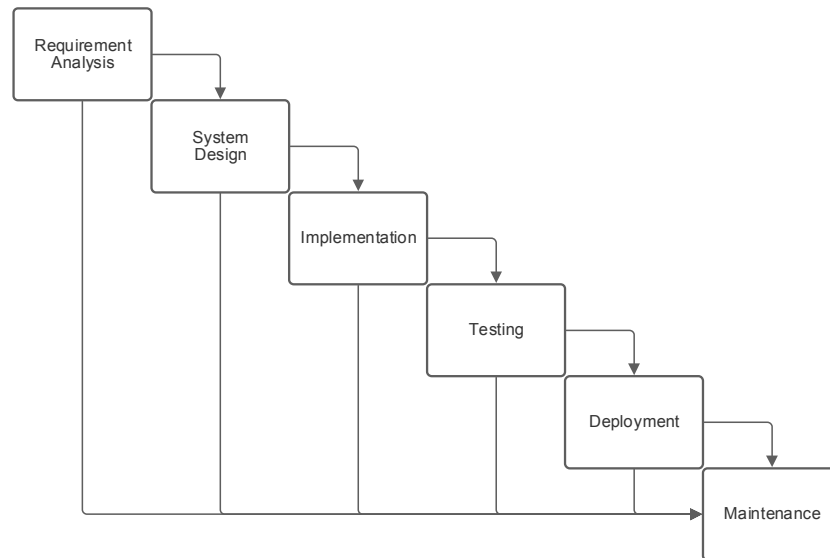


Figure 1. Waterfall model.

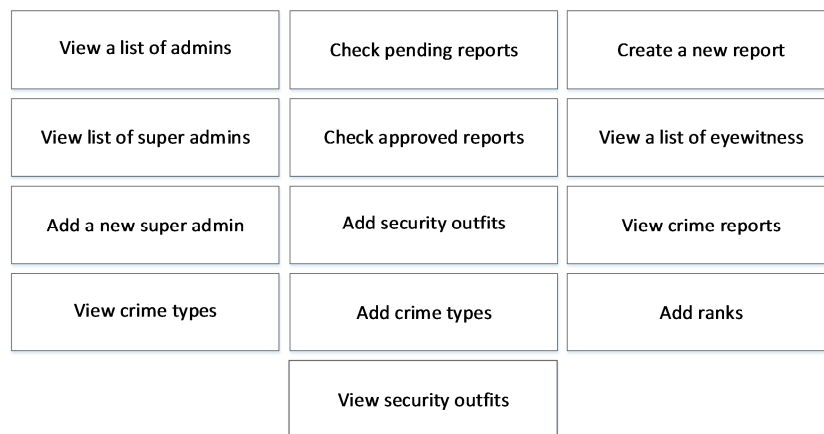


Figure 2. Major system requirements.

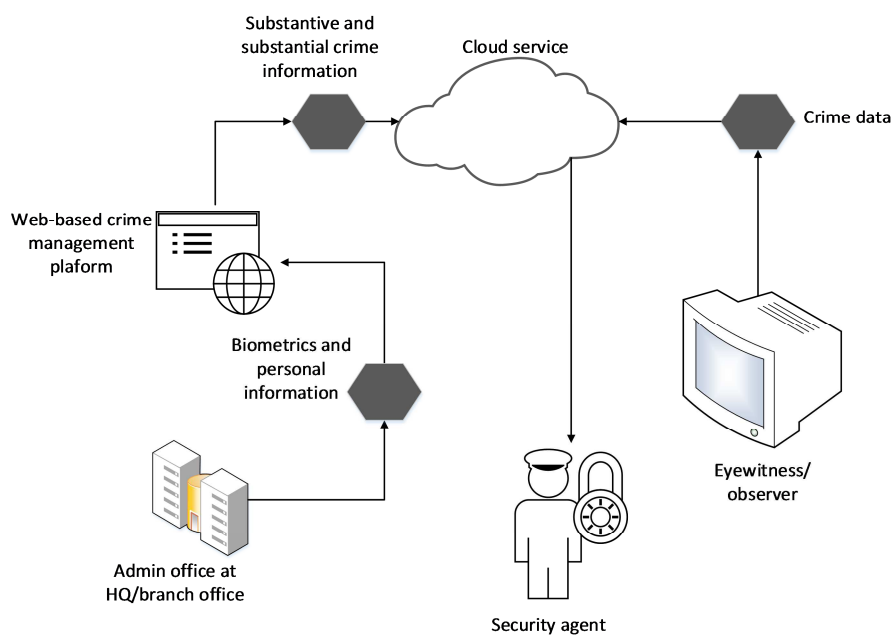


Figure 3. System architecture.

Crime will be reported by eyewitnesses/observers via a dedicated mobile application. These crime data are stored on a MongoDB database that should be hosted on a cloud service (MVPS virtual cloud service). These crime data are then accessed, filtered and managed through the web-based crime management system.

3.2. System Design

The functional modules of the web platform were designed and the information flow between subcomponents are presented in this subsection. Object-oriented programming design concepts were used in depicting the interaction between these modules. In this subsection, the UML Class Diagram and UML Sequence Diagram were used to illustrate the functional modules of the mobile app.

3.2.1. UML Class Diagram

The UML Class Diagram is a static diagram that is used to represent the static structure of an application. It provides a visualization, description and documentation of different aspects of a software system, ensuring that are convertible to executable program codes. UML class diagrams facilitate the realization of a software system based on the requirements of the user [12]. The UML class diagram shows the types being modeled within the software system. These specific types are class, interface, data type and component. The UML class diagram is characterized by some established relationships such as inheritance, associations and aggregation. The crime management was modeled using the UML class diagram shown in Figure 4.

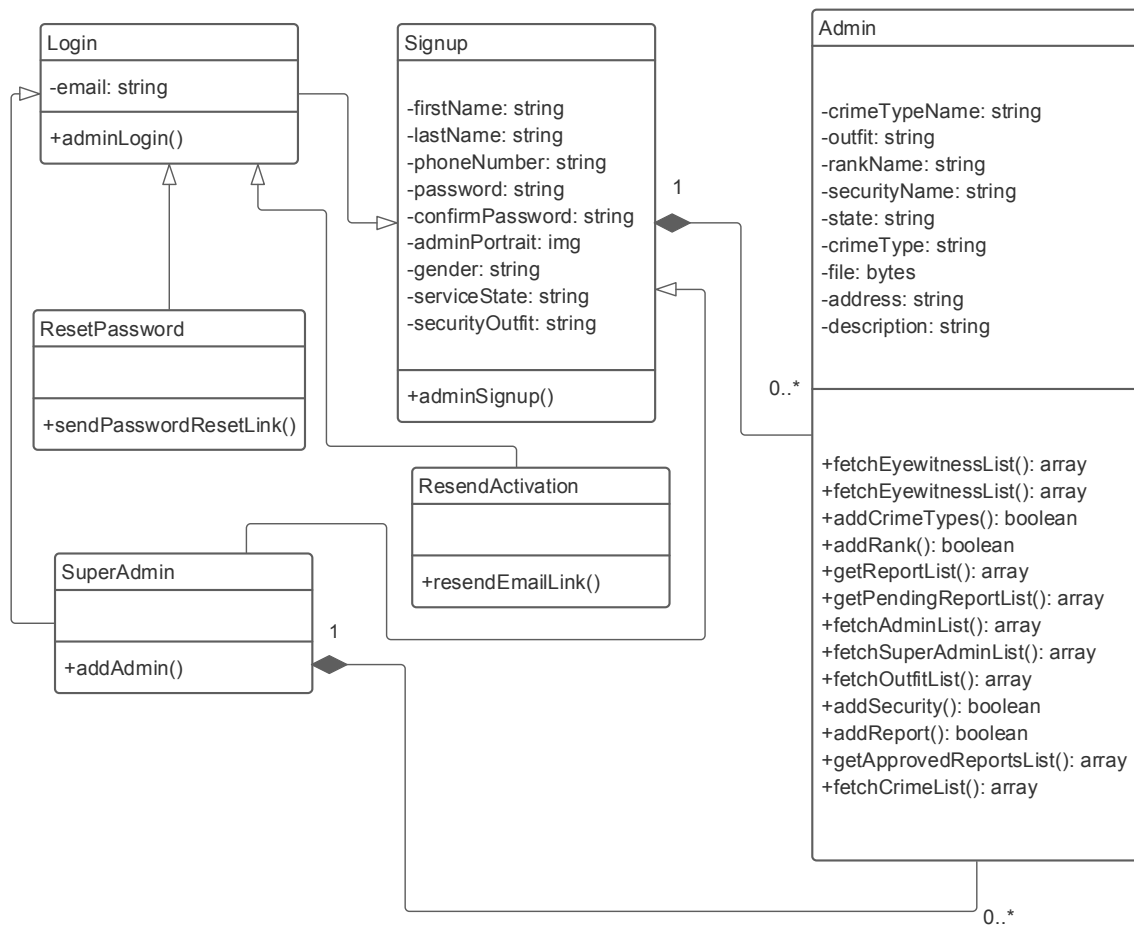


Figure 4. The UML class diagram.

From Figure 4, there are essentially six (6) classes, namely *Login*, *Signup*, *Admin*, *ResetPassword*, *SuperAdmin*, and *Resend Activation*. The *Login* class has an *email* private field and an *admin Login ()* public method. The *Login* class inherits the public properties of the *Signup* class. The *ResetPassword*, *SuperAdmin*, and *ResendActivation* classes also inherit the properties of the *Login* class. The *Signup* class has a one-to-many relationship with the *Admin* class, which implies that any

signed-up admin user can have access to the properties and methods of the *Admin* class. On a slightly high-level scale, the relationship between the *Admin* class and *Signup* class can be further interpreted as a signed-up admin user can see all the crime data reported on the platform. The *SuperAdmin* class also has a one-to-many relationship with the *Admin* class, which means that a Super Admin can either see all accredited admins on the platform and can also add new admins to the platform.

3.2.2. UML Class Diagram

The UML sequence diagram shows the interaction between the objects that make up a software system in time sequence [13]. The sequence of messages exchanged between objects needed to carry out a function are also captured in a sequence diagram. Figure 5 shows the UML sequence diagram used in this study with the accompanying objects. A new admin user registers on the system by providing his credentials. His credentials are then verified and validated before sending them to a dedicated database. If

the entire process of admin user authentication was successful, the system sends a response to the admin user that redirects him to the Login Page. In an event where validation / authentication fails, the user is redirected to the Register page to repeat the entire process of accreditation. If however, the admin user has not been activated after validation failure, an activation link is generated and sent to the user. A successfully logged-in user will now be able to access the feature-rich dashboard.

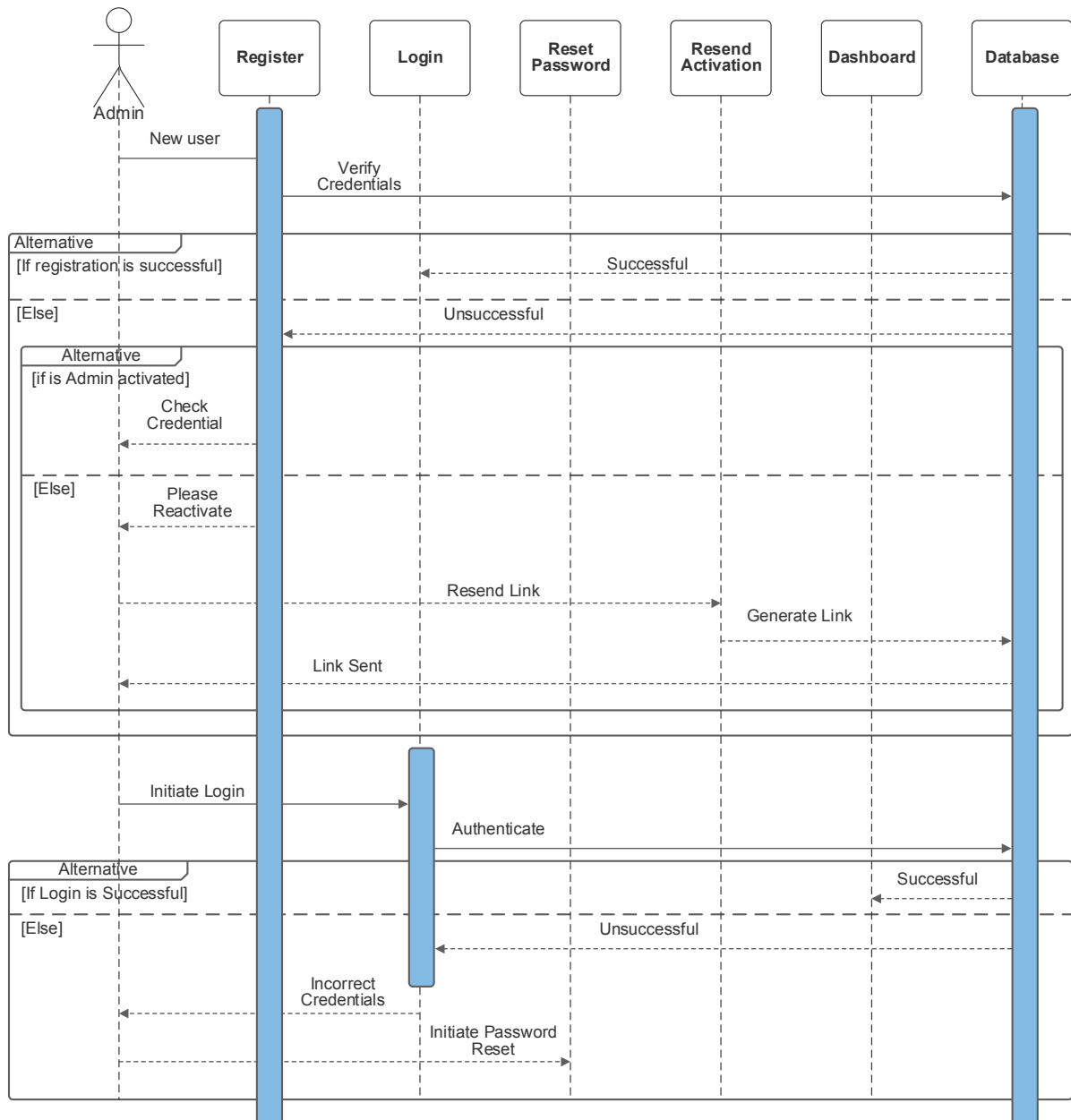


Figure 5. The UML sequence diagram.

3.3. System Implementation

The architecture of the system was implemented using IBM's three-tier enterprise application architecture as shown in Figure 6. The architecture begins with the administrative

user who interacts with the system. The main system architecture consists of the *Presentation Tier*, *Application Tier* and *Data Tier*, which follows that sequence. The following subsection discusses these layers as it relates to the system architecture.

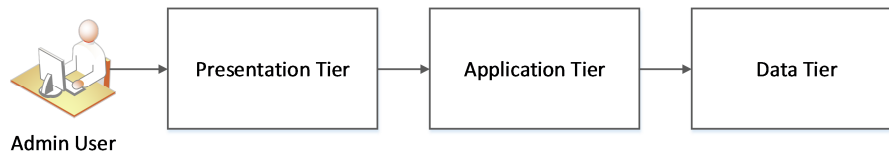


Figure 6. Three-tier Architecture for Enterprise Application [14].

3.3.1. Presentation Tier

The presentation layer defined characteristics such as user interface (UI) and user experience (UX). In this layer, users interact with the UI's graphical elements, such as buttons, labels and fields, in an intuitive manner. In addition, information is retrieved from users and information stored in a database can be rendered in this layer. In this work, frontend technologies, including HTML, CSS, Vanilla JavaScript and React, were used to design the UI of the website and provide interactivity between the admin user and the website.

3.3.2. Application Tier

The application tier is where the business logic of the website is implemented. Custom classes, packages and functionalities are implemented in the application tier. In this study, the BLOC (Business Logic Object Component) design was used as an interface between the presentation layer and business logic. This allows for smooth running of the action, allowing for scalability and maintainability. The business logic of the crime management website was implemented with Node JS and Vanilla JavaScript.

3.3.3. Data Tier

In the data tier, the application programming interfaces (APIs) provided the interaction between the website and the database. These APIs were specific to the major classes as shown in the UML class diagram. For example, the *signup* API was used to implement registration, while the *login* API was used to implement user login. Hence, the data tier enables the website to communicate with an external database that is hosted on a server on the internet.

3.4. System Testing

The performance of the developed crime management system was evaluated using alpha and beta testing to ensure that the developed system meets its specific functional requirements.

3.4.1. Alpha Testing

Alpha testing was conducted at every stage throughout the website development cycle. This was done to ensure that every module that was added to the system at each stage behaves as expected with minimal need for fine-tuning. This testing included both functional and non-functional testing to verify the performance, usability and reliability of the mobile app when it finally gets to an end user.

3.4.2. Beta Testing

Beta testing was after project completion and deployment. This test was conducted by real users to rate the performance of

the website based on certain established criteria, which includes clarity, usability, attractiveness, and responsiveness. A total of 20 neutral respondents were invited to rate the performance of the website based on these criteria. Each index was rated on a scale of 1 to 10, where '1' represents a poorly-performing index and '10' represents an excellently-performing index.

4. Results and Discussions

4.1. User Interface

4.1.1. Accreditation Pages

The website's accreditation pages (Login and Register pages) are shown in Figures 7. The register pages [shown in Figure 7 (a)] present a web form with generic input fields (such as First Name, Last Name, Phone Number, Email, Password and Confirm Password), and additional fields (such as Gender, State and Security Type). An admin officer completes these fields and submits them for validation and authentication. The Login Page [shown in Figure 7 (b)] presents a Login form with Email and Password field to be completed by an already-registered user. The Login page has other features such as Forgot Password and Resend Activation Email.

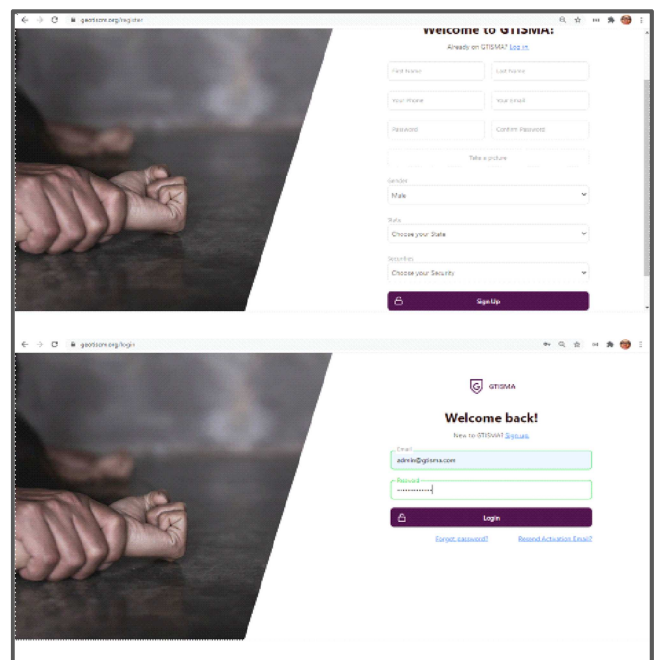


Figure 7. Accreditation Pages.

4.1.2. User Dashboard

The User Dashboard is the most important feature of the crime management system (website). An admin officer who

has been authenticated and authorized to access the system would now have access to these features. The Home Page of the dashboard gives a cursory view of the total number of reported crime activities and the total number of eyewitnesses that have reported these crime incidents (Figure 8). Other information such as the total number of approved cases and total number of specific crimes can also be seen at

a glance. The sidebar of the dashboard has other links that redirect admins officers to certain features that were highlighted when planning out the requirements of the website. Some of these features are *New Super Admin* (Figure 9), *View Eyewitness* (Figure 10), *View Reports* (Figure 11) and *View Report Details* (Figure 12).

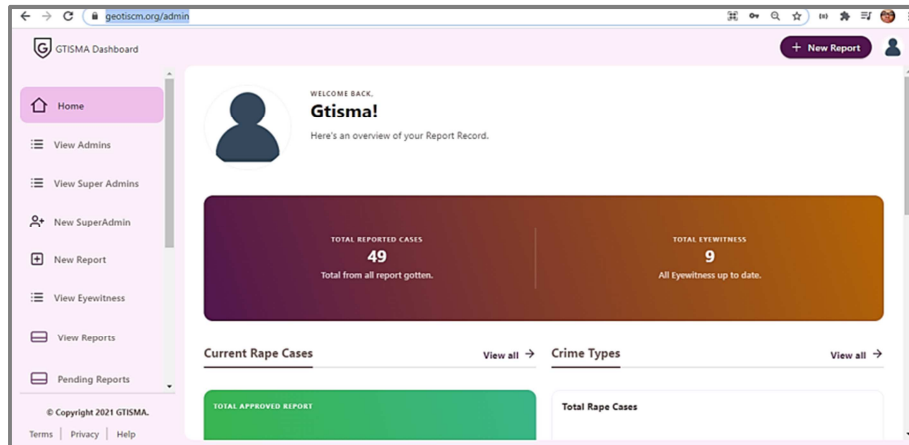


Figure 8. User Dashboard.

The screenshot shows the 'Add New SuperAdmin' form. It includes input fields for 'First Name', 'Last Name', 'Your Phone', 'Your Email', 'Password', and 'Confirm Password'. There are also dropdown menus for 'State' (currently set to 'Abia') and 'Gender' (currently set to 'Male'). An 'Add Admin' button is at the bottom.

Figure 9. Add new super admin.

The screenshot shows the 'Eyewitness List' table. The table has the following columns: FIRSTNAME, LASTNAME, EMAIL, PHONE, GENDER, REGISTERED WITH, and DATE REGISTERED. The data rows are as follows:

FIRSTNAME	LASTNAME	EMAIL	PHONE	GENDER	REGISTERED WITH	DATE REGISTERED
Francis	Akinluyi	fakinluyi@gmail.com			facebook	2021-02-23 00:48:38
Joseph	Minguel	clashofdures@gmail.com			facebook	2021-02-16 23:57:25
Akin	Bello	belloakin24@gmail.com			facebook	2021-02-10 04:44:18
Sandra	Ezeibe	ezeibesandra-1@gmail.com			google	2021-01-18 03:29:05
Sandra	Ezeibe	ezeibesandra@gmail.com			facebook	2021-01-18 03:03:16
Victor	Iheakababa	iheakababavictor@gmail.com			facebook	2021-01-15 02:47:52
Olebu	Chinedu	chineduolebu2021@gmail.com	07061231479	Male	regular	2021-01-14 04:48:09
Abimbola	Iremobayo	abimbolairemobayo@gmail.com			facebook	2021-01-14 04:37:31
Chinedu	Olebu	cgolebu@gmail.com			facebook	2021-01-13 11:08:29

Figure 10. View list of eyewitnesses.

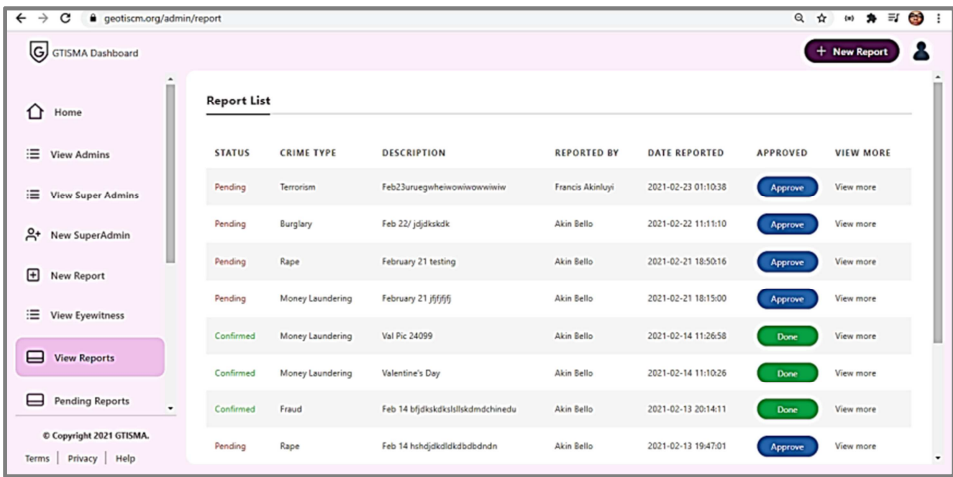
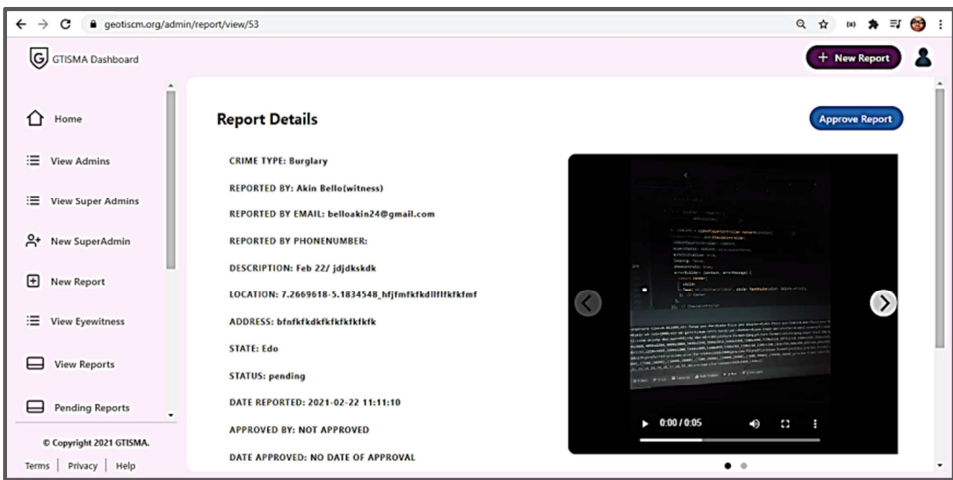
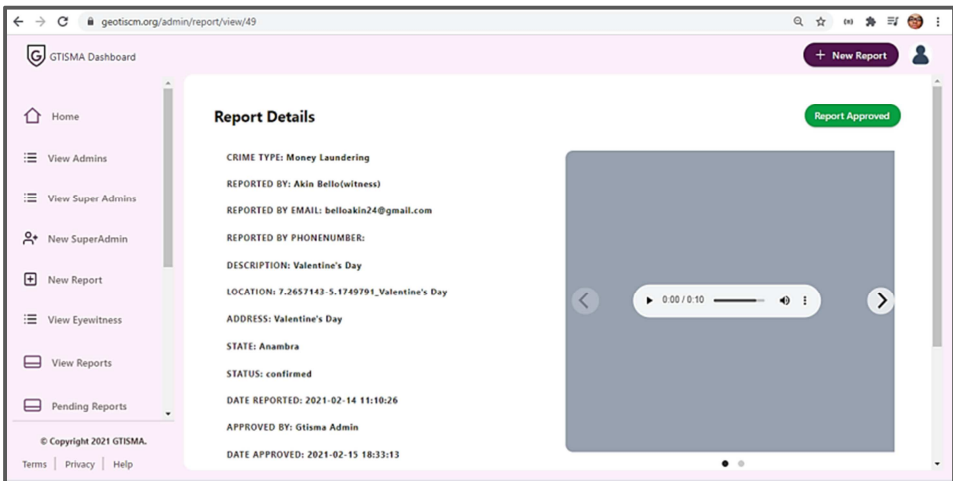


Figure 11. View List of Reports.



(a)



(b)

Figure 12. View report details.

The View Report Details feature shows complete information about a particular crime incident reported. It captures information such as crime type, reporter's name, location, address and description. The reports could be in the form of video, audio and simple texts based on the information provided by the eyewitness through a dedicated

reporting mobile application. For example, in Figure 11 (a), the reporter reported a crime incident by recording the crime activity in video format, while Figure 11 (b) shows that the reported recorded the audio feeds from the crime activity. Hence a video player was used to render the recorded video, while an audio player was used to render

the recorded audio.

4.2. Performance Evaluation

The performance of the website was evaluated using two major tools, namely Google Chrome built-in tool: Lighthouse and Beta Testing. The result of using Lighthouse to test the performance of the website when live is summarized in Figure 13.

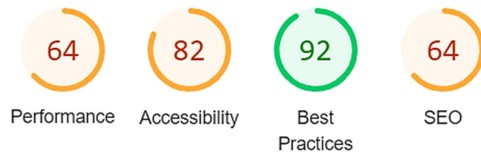


Figure 13. Lighthouse performance result.

One of the predominant metrics of the performance index is the Time to Interactive (Time to interactive is the amount of time it takes for the page to become fully interactive). It took the page about 2.2s to become fully active. This time is quite satisfactory, hence there is not much delay when the page is loaded. The second index (Accessibility) emphasizes the ease of use, especially potential users that could use assistive devices to access the website. The percentage accessibility of the website is 82% which means that fonts

are legible and there are good contrasts between different colors of fonts and backgrounds. The third index (Best practices) has a percentage rating of 92%. This implies that the site is protected against XSS attacks and has much lower security vulnerabilities and avoids using deprecated APIs. The last performance index [Search Engine Optimization (SEO)] which has a rating of 64% ensures that the web page is following basic search engine optimization advice. The result from Lighthouse analysis shows that the webpage behaves as expected and there is not much overhead on the performance of the website. It should be noted that the performance report of Lighthouse should not be considered as the 'silver bullet' for measuring the performance of a website.

The evaluation results of beta testing were obtained and presented in Table 1. The table shows the percentage beta performance ratings obtained organically from real users. From the table, *User Interface* has a percentage rating of 96%, the web site's *Functionality* has a performance rating of 94%, *Clarity* of functionality has a rating of 92% and *Maintainability* has a rating of 91%. This shows that the developed crime management system performs satisfactorily and effectively serves its purpose.

Table 1. Developed crime management system beta testing result.

User Index	User Interface	Functionality	Clarity	Maintainability
1	10	9	8	8
2	10	9	10	10
3	8	10	10	9
4	10	10	10	9
5	10	8	9	9
6	10	10	8	8
7	10	10	10	9
8	8	9	10	9
9	10	10	8	10
10	10	9	9	10
Average Rating	9.6	9.4	9.2	9.1
Percentage Rating	96%	94%	92%	91%

5. Conclusion

A crime management system named GEOTISCM was developed for southwestern Nigeria. GEOTISCM is a transparent, accountable, and all-inclusive platform for crime management and monitoring. It factors in important features that enables it to be easily adoptable by different security architectures in Southwestern Nigeria. The crime management platform implements separation of concerns in a unique way by creatively separating crime reporting from management of crime data. The waterfall software development technique was used in developing the system. The core requirements of the system were highlighted and designed using the Unified Modeling Language (UML). The system was then implemented using IBM's three-tier architecture. The entire development process of the system culminated in Testing,

deployment to cloud platform and iterative maintenance. The resulting website can now be accessed via a domain name. The performance of the crime management website was determined using Google Chrome's built-in tool called Lighthouse and the user experience (UX) of the website was measured by conducting beta tests with a total of 10 respondents. The Lighthouse showed that the website performed satisfactorily, thus, satisfying the core requirements of the system. Also, analysis of feedback from respondents showed that potential users would not face any difficulty in using the crime management system.

Funding

The work is sponsored by TETFund NRF 2019 with reference number TETFund/DR&D/CE/NRF/HSS/01/VOL1. The authors appreciate the contributions of the entire GTIS

FUTA TETFund Research Team in the evaluation of the developed Crime Management System.

References

- [1] Erling S. (2015). Crime Management in Nigeria: A Sociological Analysis. *Multi-Disciplinary Journal of Research and Development Perspective*. Vol 4, No 2, pp (93-96).
- [2] Jeffrey O. A., Chjioke E. C., and Daniel I. Y. (2022). An Intelligent Crime Management System for Lafia Metropolis. *FUDMA Journal of Sciences (FJS)*. Vol 6, No 3, pp (139-150).
- [3] Archana M., Durga S. and Saveetha, K. (2019). Online Crime Management System. *International Journal of Advanced Networking and Application*. Pp (297-299).
- [4] Tomas U. G., John, S. C., Ronalyn D. C., and Jeromme G. P. (2019). Development of an Online Crime Management and Reporting System. *World Scientific News, An International Scientific Journal*. Pp (164-180).
- [5] Saqib N., Junaid G., Asif S., and Muhammed A. (2019). Online Crime Records Management System: A Case of Paskistan. *Journal of Information Engineering and Applications*. Vol 9, No 6, pp (11-20).
- [6] Eyal K. (2021). Top 7 PHP Security Issues and Vulnerabilities. <https://spectralops.io/blog/top-7-php-security-issues-and-vulnerabilities/>
- [7] Manisha G. I., Snehel W. D., Roshi, S. R., Kavita B. R., and Chavhan K. I. (2021). Crime Report Management System. *International Journal of Modernization in Engineering Technology*. Vol 03, Issue 7 pp (760-764).
- [8] Oludele, A., Onuiri, E., Olaore, O, Sowunmi, O, & Ugo-Ezeaba, A. (2015). A real-time crime records management system for National Security Agencies. *European Journal of Computer Science and Information Technology*, 3 (2), 1-12.
- [9] British Council (2016). Crime Management System. www.j4a-nigeria.org. (1-5).
- [10] Corrales E. (2022). Waterfall Software Development Methodology? <https://www.developer.com/project-management/sdlc-waterfall-model/>
- [11] Martin F. (2019). Waterfall Process. <https://martinfowler.com/bliki/WaterfallProcess.html>.
- [12] Nikiforova, O., Sejans, J., and Cernickins, A. (2011). Role of UML class Diagram in Object-Oriented Software Development. *Scientific Journal of Riga Technical University*, 44, 66-74.
- [13] Al-Fedaghi, S. (2021). UML Sequence Diagram: An Alternative Model. *International Journal of Advanced Computing Science and Applications*, 12 (5), 635-645.
- [14] IBM Cloud Education. (2020). Three-Tier Architecture. Retrieved July 17, 2021, from <https://www.ibm.com/cloud/learn/three-tier-architecture#:~:text=Three%2Dtier%20architecture%20is%20a,associated%20with%20the%20application%20is>.