

Design And Implementation Of Web Application With Google Map Location Visualization For Lecture Timetable In University Of Uyo

Kalu Constance¹

Department of Electrical/Electronic and Computer Engineering, University of Uyo, Akwa Ibom, Nigeria
constance.kalu@yahoo.com

Jude Ibanga²

Department of Electrical/Electronic and Computer Engineering, University of Uyo, Akwa Ibom, Nigeria

Nseabasi Peter Esien³

Department of Computer Education,
University of Uyo, Akwa Ibom, Nigeria

Abstract— The paper presented a web-based lecture timetable management system for University of Uyo (UNIUYO). The web application uses Google Maps Application Programming Interface (API) to interface the timetable with Google map and thereby allows the lecture venues to be mapped and visualized online on the Google map. The study is motivated by the peculiar problems that are associated with timetable management in UNIUYO. Notably, UNIUYO has about four different campuses that are few kilometers apart from one another. Some students offer course combinations that require the students to have their lectures in different campuses of UNIUYO. Accordingly, the major focus of the timetable web application is to automate the timetable design process, eliminate or at least minimize conflicts in the timetable, provide a platform for prompt resolution of problems identified by the users and to make it easier for users to locate the lecture venues in the various campuses of university of Uyo. The Waterfall software development method is used in the web application development. The tools used for the system implementation include: Macromedia Dreamweaver and Cascaded Style Sheet (CSS), MYSQL Database Management System, PHP for the server-side scripting language and Apache for the Web server. Google Maps Application Programming Interface (API) is embedded in the system to allow communication between the application on the web-server and the remote Google geo-server. Sample lecture timetable data from the faculty of Engineering University of Uyo were used to demonstrate the application of the web-based time table management system.

Keywords — *Timetable, Map Mashup, Web Application, Spatiotemporal, Software Development Methodology*

I. INTRODUCTION

Over the years, lecture timetables have been indispensable tools for the efficient running of schools,

especially, higher institutions where there are different disciplines with many courses that need to be assigned time and venue, even in the face of several constraints [1,2,3,4]. Interestingly, the emergence of Information and Communication Technology (ICT) has brought about computerized methods for preparing and publishing time tables [5,6,7,8,9,10,11]. In addition, the emergence of map mashup application has introduced another dimension to the timetable software system. It is now possible to include the spatial information about the lecture venues which can enable map-based visualization and tracking of the lecture venues [12, 13,14].

Consequently, in this paper, the design and implementation of web application with Google map mashup application for lecture timetable in University of Uyo (UNIUYO) are presented. The major problem is that UNIUYO has about four different campuses that are few kilometers apart from one another. Some students offer course combinations that require the students to have their lectures in different campuses of UNIUYO. Again, some courses at the first and second year are offered by all the disciplines in the university, as such, time table for such courses are centrally prepared before each discipline can prepare their own time tables. Most often, there are clashes in lecture venues and times among the various disciplines in the university. Besides, in some cases, those students that need to travel to another campus for their lecture do not have enough time in between the lecture times for such trip. Unfortunately, the manual approach currently used for lecture timetable does not ensure prompt resolution of conflict in the time table. Furthermore, some students who are not familiar with some of the campuses spend most of their times searching for the lecture venues. Also, there are some visiting fellows or part-time staff and students who are not familiar with different facilities in the university. In view of these problems, a web-based time table with Google map enabled mapping and visualization of the lecture venues is developed for UNIUYO. The web application enables the university to capture all relevant information concerning the timetable preparation; among the required information are the details of the courses, the lecture venues, the student's course registration schedules and the

lecturer's course allocations. It also captures the geographic coordinates of the lecture venues. The web application then uses its algorithms to schedule lectures for the courses in the available venues in such a way that the conflict in time or venues are avoided. The web application also interfaces with Google Map web server through the Google Maps Application Programming Interface (API) which enables the system to communicate between the web-server and the Google Map web server [15, 16,17,18, 19]. Eventually, the web server uses the local data captured from UNIUYO to visualize the lecture venues on Google map. It also allows users to use other Google map features to track the lecture venue in real-time.

II. THE SUMMARY OF THE PROCEDURE USED TO ADDRESS THE PROBLEMS IDENTIFIED IN THE EXISTING UNIVERSITY OF UYO TIMETABLE MANAGEMENT SYSTEM

The major focus of the timetable web application is to automate the timetable design process, eliminate or at least minimize conflicts in the timetable, provide a platform for prompt resolution of problems identified by the users and to make it easier for users to locate the lecture venues in the various campuses of university of Uyo. The overall timetable web application is managed by an administrative team (admin) that has access to the admin module (Figure1). The admin manages the web application hosting plan, ensures that the users operate within their user privilege specifications, attends to the complaints posted by users and also responds to those complaints. In order to minimize conflicts in the timetable, the web application provides a centralized procedure for creating, updating and validating the timetable records. The centralized procedure is summarized as follows;

- i. At the beginning of each academic session, the timetable web application admin unit updates the database with detailed information on all the courses offered in the university, the lecture venues in all the campuses and their geo-coordinates, the lecturer's course allocation and other information like holidays and lecture free days.
- ii. Since students' course registration details for the current semester may not be ready as at the time the timetable is prepared, the previous year's course registration details and results are used to generate the probable course registration details for the current semester. This approach may lead to few conflicts in the timetable. However, the complaint module in the web application is used to ensure that the users' complaints are received and treated promptly.
- iii. The information on the lecture venues capacity and the class size of each of the courses are updated and the lecture venue size and course class size are matched. This application will develop a table that generates a list of various lecture venues that can accommodate a given course based on the lecture venue size and course class size compatibility. The guiding principle for the allocation is that the venue size, in terms of number of people the venue can accommodate, must be greater or equal to the course class size.
- iv. Next, the courses are assigned lecture venues starting from the first year courses, followed by the second year, and then up till the final year courses. In the time table, the postgraduate students are not considered since the postgraduate students have their separate venues that are different from the ones used by the undergraduate students. The reason why the allocation of time and venues start with the lower levels is that, one, some courses at the lower levels (such a General Studies, GST courses) are offered by all the students in that level. So, majority of the lecture venues will be occupied when these courses appear on the timetable. Two, some students at the higher level s have some lower level courses that they failed and which they need to re-take. So, by assigning the lower level courses first, the conflict in the timetable due to carryover courses are eliminated or minimized.
- v. After the year one courses are allocated, the year two courses are allocated. When a course in year two is assigned a venue and time, the system validates the allocation with the central database. The validation checks that there is no conflict in venue, in time, in any carryover student's time, there is sufficient time for those students and staff that will travel from one campus to the other campus to attend the class. All the information required for such decisions are stored in the centralized timetable database management system. Similar procedure is used to allocate venue and time to the courses for the other higher level courses.
- vi. Each academic discipline in the university must submit the detail information to the central timetable admin unit which uses the information to create and update the web-based lecture timetable system. After the

timetable is created and uploaded to the web server, the students and staff can view the timetable and they can also login with their account detail and they can extract their personal timetable from the online timetable.

- vii. Any complaint on the timetable can be posted to the admin. Again, it is only registered users with username and password that can submit complaints. The timetable admin team will respond to the complaint and also take steps to resolve it.

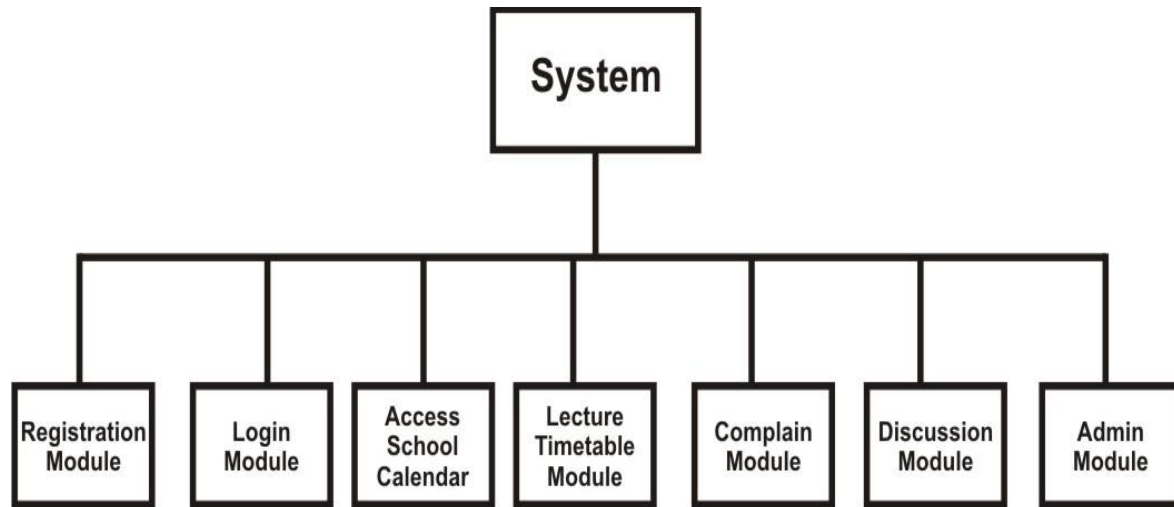


Figure 1 Functional Decomposition Of The System

III. SYSTEM DESIGN

In this paper, the modified Waterfall software development methodology (Figure 2) is adopted for the development of the web-based lecture timetable management system. The activities involved in the Waterfall model include requirement engineering, planning, design, coding, testing, integration and deployment. Requirement engineering is a critical phase of the Waterfall model as such a robust requirement process model (Figure 3) is employed and it consists of the following activities: requirement elicitation, requirement analysis, requirement validation, and requirement specification. Requirement elicitation consists of actions and strategies employed by the system developers in order to understand the users' viewpoint on the system. In this work, the requirement elicitation techniques (Figure 4) used are interview, meeting with the stakeholders, brainstorming, ethnography and the literature review.

In the system planning and design, the requirements that are gathered are used to identify various system modules and then decompose the system into the identified modules (Figure 4). Each of the modules is further decomposed into submodules and appropriate flowcharts are developed for each of the submodules. Sample flowcharts are shown in Figure 5 for the registration module and Figure 6 for the login module. Further functional decomposition of the admin module is presented in Figure 7 while that of the Complaint module is presented in Figure 8. The usecase diagram for the system is presented in Figure 9. The database is implemented with MYSQL and it contains the following tables: venue (Table 1), faculties (Table 2), department (Table 3), days (Table 4), studentreg (Table 5), course of study (Table 6), allocate venues (Table 7) and login (Table 8). The information architecture of the web pages is shown in Figure 10. The information architecture used wireframe diagram to show the layout of the contents on the web pages.

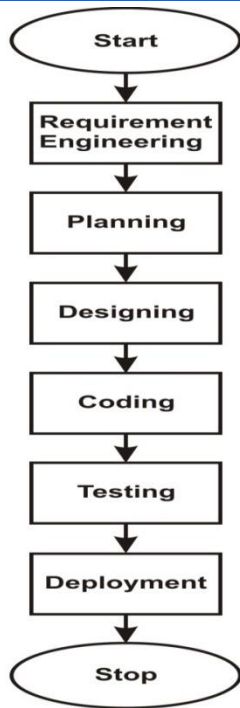


Figure 2 The Modified Waterfall Software Development Methodology

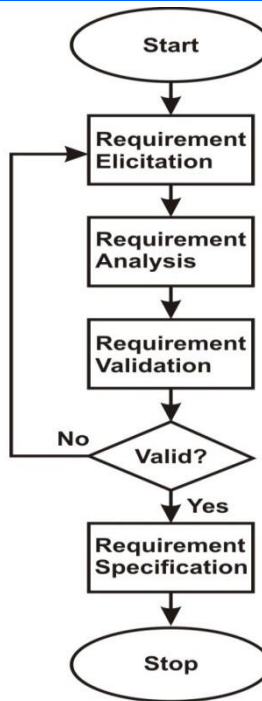


Figure 3 The Requirement Process Model

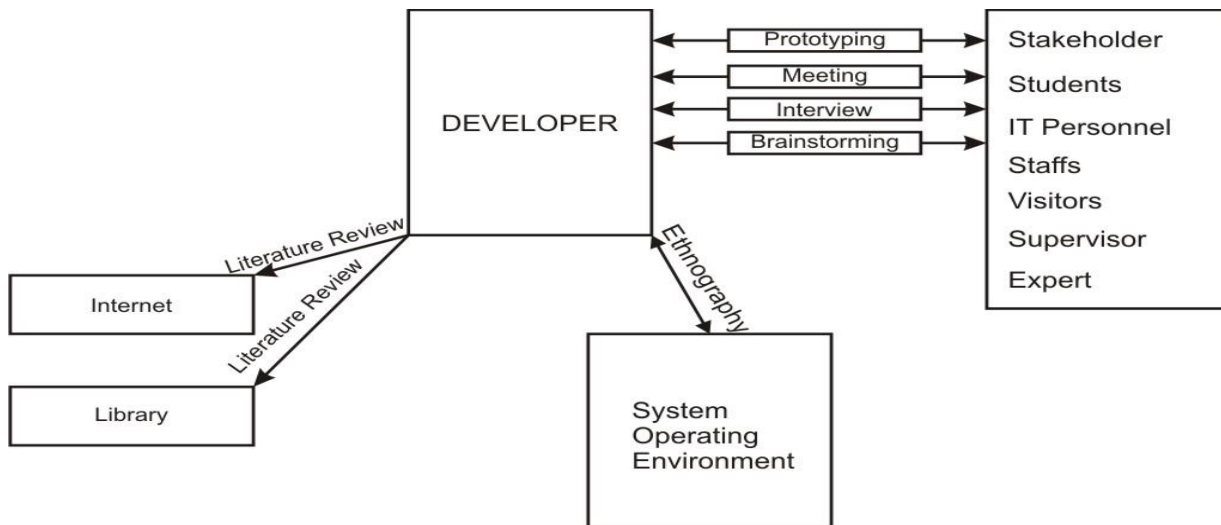


Figure 4 The Requirement Elicitation Techniques

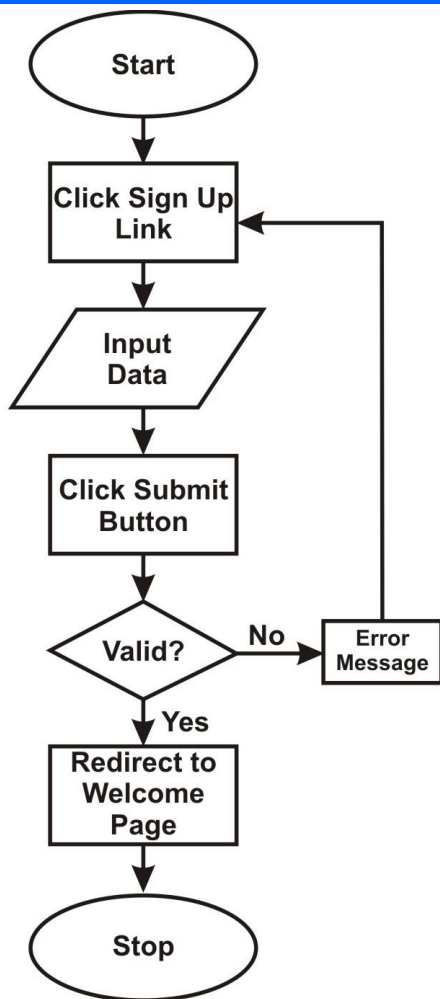


Figure 5 Flowchart for the registration module

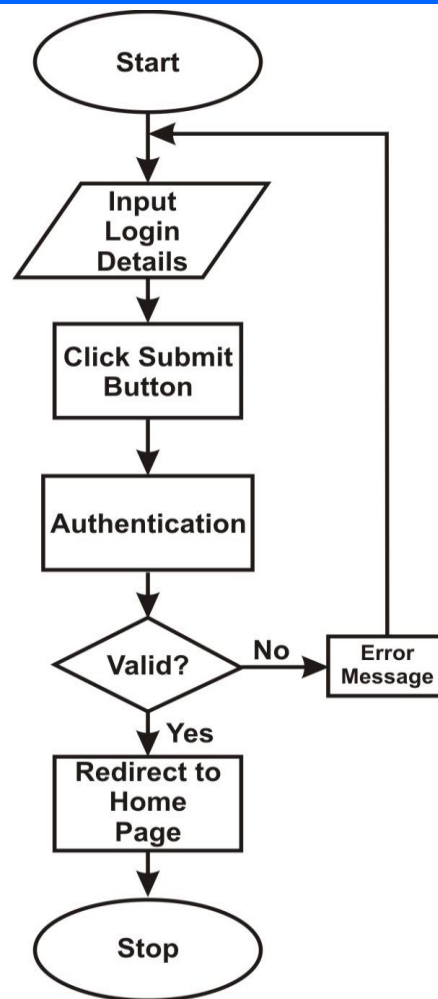


Figure 6 Flowchart for the Login module

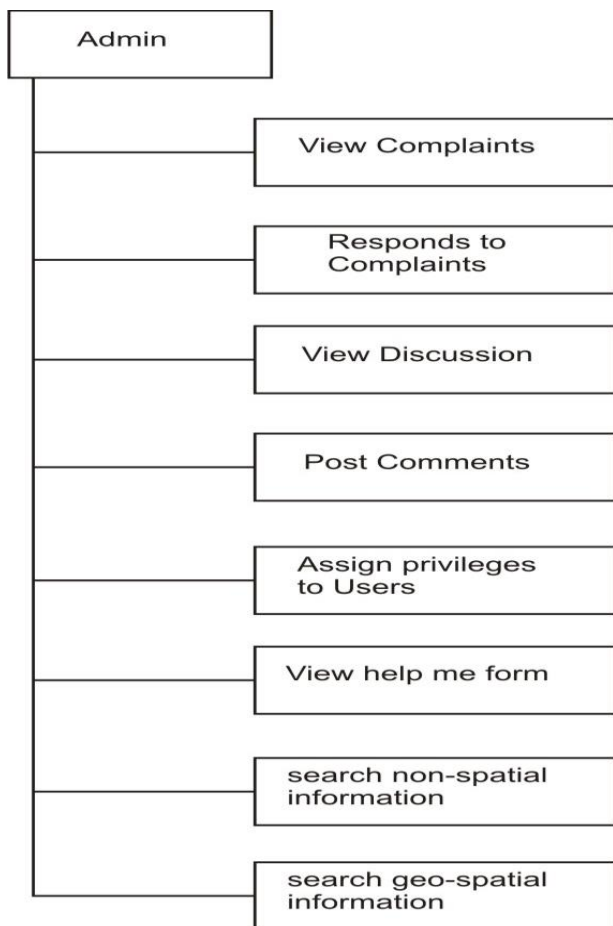


Figure 7 Functional Decomposition of The Admin Module

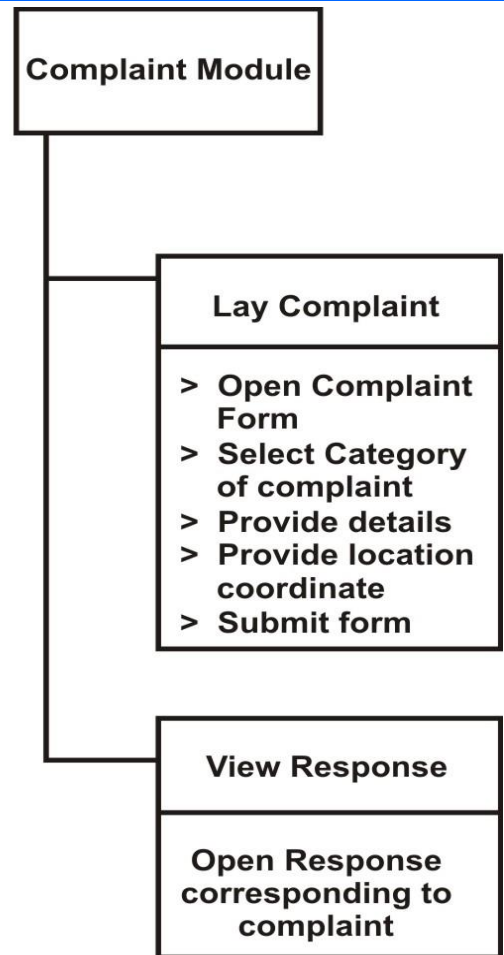


Figure 8 Functional Decomposition of The Complaint Module

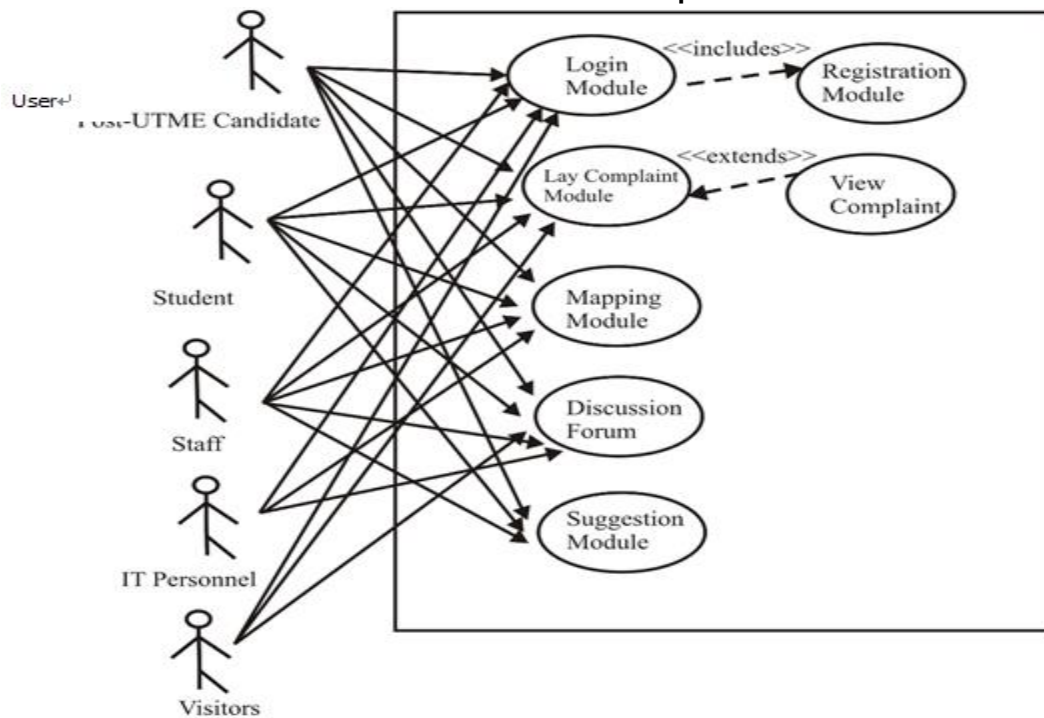


Figure 9 The Used Case Diagram for the system

Table 1: Table structure for table: venue.

Column	Type	Null
id	int(11)	No
venue_code	varchar(10)	No
venue_name	varchar(255)	No
venue_description	Text	No
lat	varchar(10)	No
longitude	varchar(10)	No
picture	varchar(100)	No

Table 2: Table structure for table: faculties .

Column	Type	Null
faculties_id	int(100)	No
faculties_name	varchar(255)	No
faculty_code	varchar(30)	No

Table 3: Table structure for table: departments.

Column	Type	Null
departments_id	int(100)	No
departments_name	varchar(255)	No
fac_id	varchar(200)	No
departments_code	varchar(100)	No

Table 4: Table structure for table: days.

Column	Type	Null
id	int(11)	No
day_code	varchar(25)	No
day_name	varchar(25)	No

Table 5: Table structure for table: studentreg.

Column	Type	Null
id	int(11)	No
surname	varchar(89)	No

othernames	varchar(89)	No
lga	varchar(89)	No
level	varchar(150)	No
dob	varchar(25)	No
sex	varchar(10)	No
course	int(11)	No
dept	varchar(150)	No
address	Text	No
nationality	varchar(25)	No
religion	varchar(25)	No
regno	varchar(25)	No
state	varchar(25)	No
email	varchar(89)	No
gsm	varchar(89)	No
image	varchar(89)	No
password	varchar(89)	No

Table 6: Table structure for table: courseofstudy.

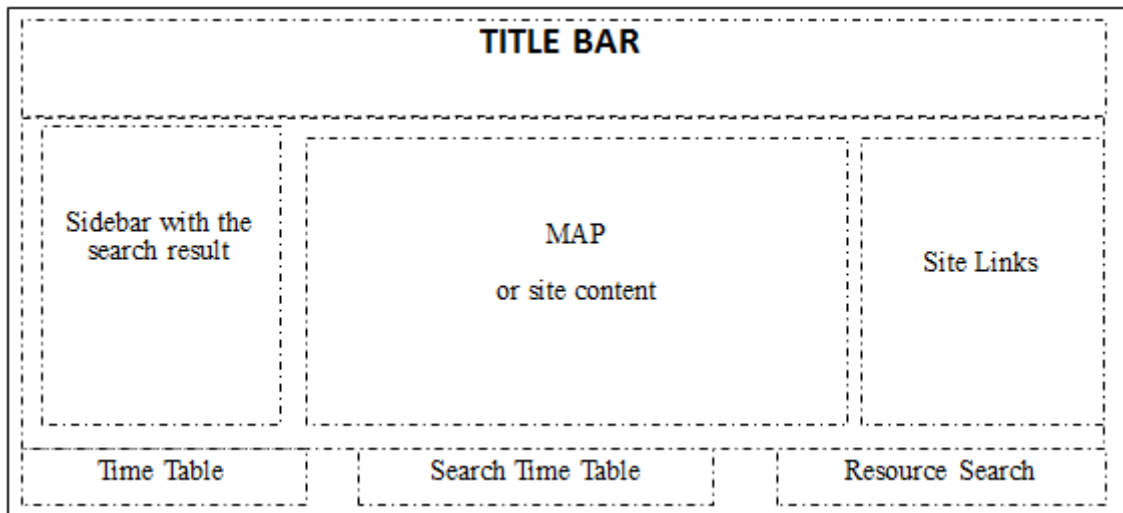
Column	Type	Null
study_id	int(11)	No
dept_id	int(11)	No
study_name	varchar(100)	No
duration	varchar(15)	No
prog_id	int(11)	No

Table 7: Table structure for table: allocate venues.

Column	Type	Null
id	int(11)	No
cstudy	int(11)	No
venue	int(11)	No
days	int(11)	No
times	int(11)	No
years	varchar(5)	No
level	int(11)	No
ccode	int(11)	No

Table 8: Table structure for table: login.

Column	Type	Null
id	int(11)	No
staffid	varchar(25)	Yes
login	varchar(50)	Yes
password	varchar(50)	Yes
fullnames	varchar(89)	No
address	varchar(150)	No
phoneno	varchar(25)	No
lastLogin	varchar(25)	Yes
status	varchar(25)	Yes

**Figure 10: Wire Frame Diagram of the Web pages**

IV. SYSTEM IMPLEMENTATION

The web application is implemented using Hypertext Markup Language (HTML) and PHP (HyperText Pre-processor) server-side scripting language, MYSQL database management system, and hosted locally using WAMP server that consists of Windows operating system, Apache web server, MYSQL, and PHP. Adobe Macromedia Dream Weaver is used for the front-end design of the web application and for editing the HTML and PHP codes. Google map API is used to interface the web application to the Google map web server. Other tools used include a GPS tool which is used for generating the coordinates, (longitudes and latitudes) of each lecture venue and a camera for taking live pictures of the lecture venues. The screenshot of UNIUYO main campus with some lecture venues in faculty of Engineering mapped on Google map is shown in Figure 11. The sign-up form

or registration form for new students is shown in Figure 12. All fields are mandatory in the form and once a student successfully registers, he/she can then login with the login details (username and password), as shown in Figure 13. Once a registered member logs into the system, the member homepage (Figure 14) opens. This shows the user details and the navigation links to access the functionalities in the system. On the student page, the student can access and check his profile, register subjects and view lecture venues. The screenshot for the lecture venues is shown in Figure 15. The system admin can add new venues and also allocate courses to venues, as shown in Figure 16. A student can view his/her personal timetable and courses that are applicable to him on the timetable. The screenshot for the timetable search webpage is shown in Figure 17. The screenshot for a sample search result is shown in Figure 18.

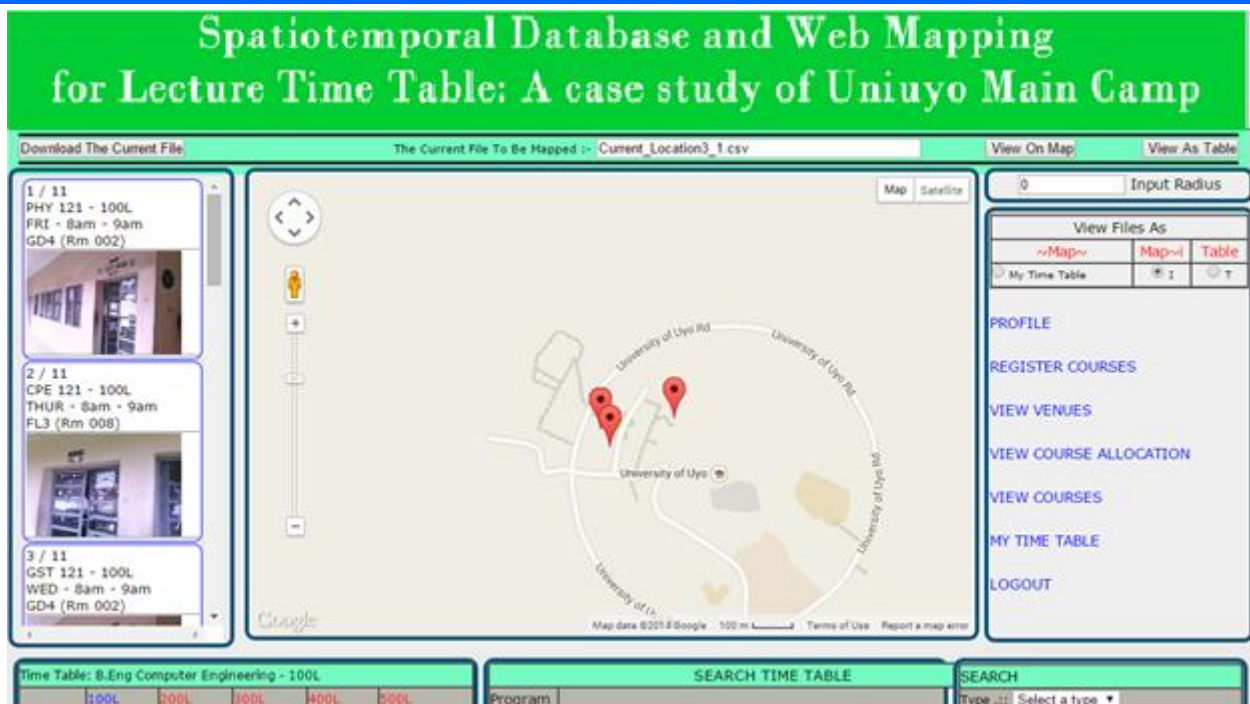


Figure 11: The UNIUYO main campus with some lecture venues in faculty of Engineering on Google map

Figure 12: The signup page with the login link.

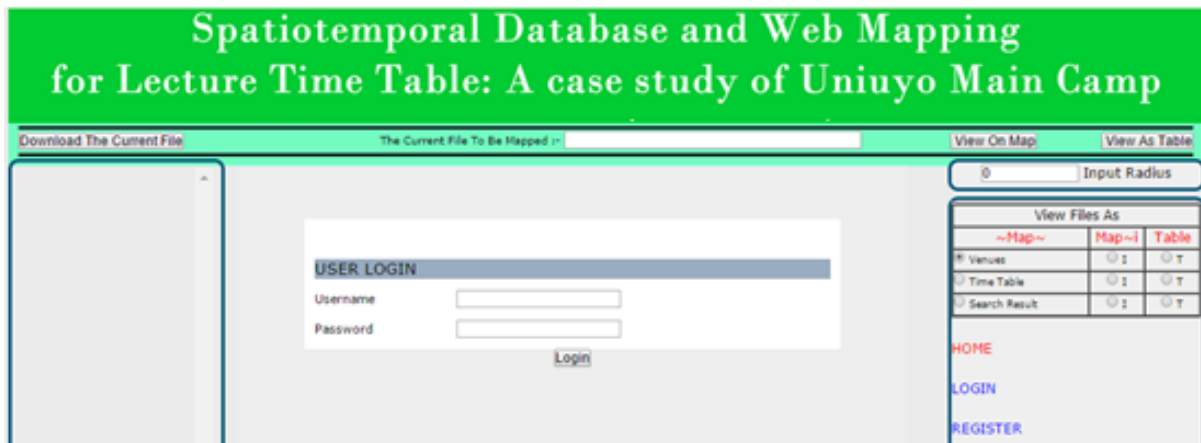


Figure 13: The signup page for new students and login for registered students.

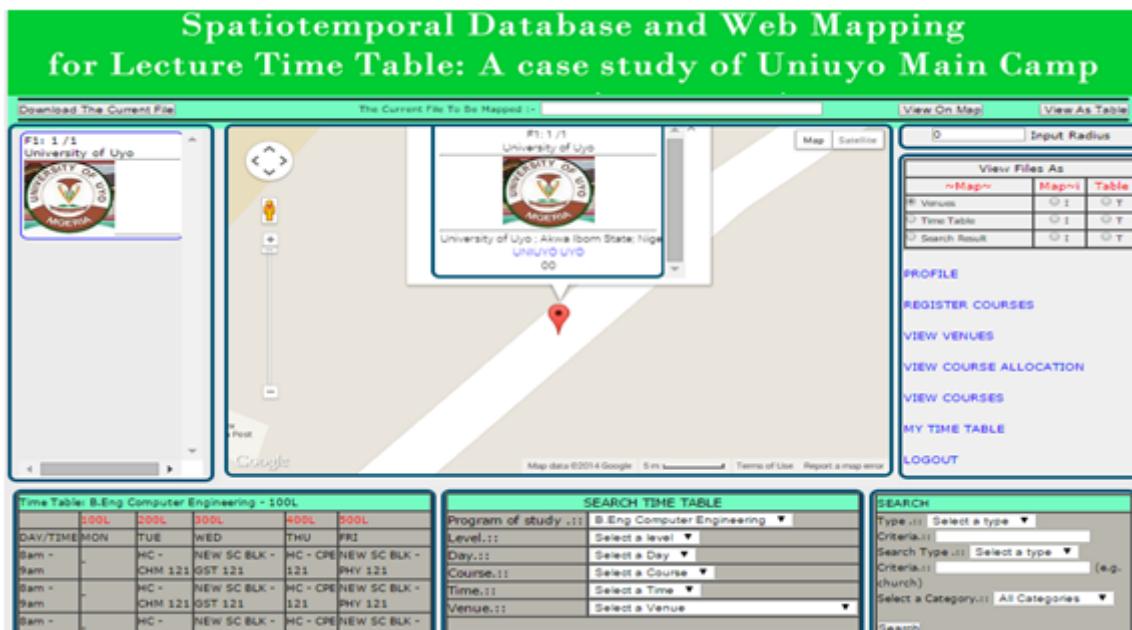


Figure 14: The Student homepage.

Download The Current File The Current File To Be Mapped :- View On Map View As Table

0 Input Radius

Welcome EFFIONG ASUQUO

VENUE

S/N	Venue	Code	Description	Latitude	Longitude
1	Agric & Food Engineering Lab II (Rm 040)	AGFLAB2	Behind PTDF Auditorium	5.041908	7.975867
2	Agric Engineering Lab (Rm 017)	AGRLAB	Opposite Engineering Drawing Studio	5.041933	7.97574
3	Chemical Engineering Lab (Rm 020)	CHELAB	Opposite Engineering Drawing Studio	5.04198	7.975883
4	Chemical Reaction & Biochemical Engr Lab	CRLAB	Beside VC's Office	5.041402	7.975915
5	Chemical/Biochemical Reaction Laboratory	CBLAB	Beside VC's Office	5.036112	7.976151
6	Chemical/Petroleum Lecture Hall	CPHALL	Beside VC's Office	5.041237	7.975753
7	Civil Engineering Lab (Rm 005)	CVLAB	Opposite Engineering Drawing Studio	5.041585	7.975098
8	Computer Engineering Lab (Rm 008)	CELAB	Opposite Engineering Drawing Studio	5.04171	7.975057

View Files As

~Map~	Map~i	Table
<input checked="" type="radio"/> Venues	<input type="radio"/> I	<input type="radio"/> T
<input type="radio"/> Time Table	<input type="radio"/> I	<input type="radio"/> T
<input type="radio"/> Search Result	<input type="radio"/> I	<input type="radio"/> T

PROFILE

REGISTER COURSES

VIEW VENUES

VIEW COURSE ALLOCATION

VIEW COURSES

MY TIME TABLE

LOGOUT

Figure 15: Venues in the system.

Download The Current File The Current File To Be Mapped :- View On Map View As Table

0 Input Radius

COURSE ALLOCATION TO VENUE

S/N	CourseCode	Venue	Day	Time	Course of Study	Level
1	MTH 121	FL3 (Rm 008)	MON	11am - 12pm	B.Eng Computer Engineering	100
2	MTH 121	FL3 (Rm 008)	MON	12pm - 1pm	B.Eng Computer Engineering	100
3	GST 221	FL3 (Rm 008)	MON	8am - 9am	B.Eng Computer Engineering	200
4	GST 221	GD3 (Rm 001)	MON	9am - 10am	B.Eng Computer Engineering	200
5	GRE 222	Engineering Workshop	MON	4pm - 5pm	B.Eng Computer Engineering	200
6	GRE 321	Engineering Drawing Studio	MON	8am - 9am	B.Eng Computer Engineering	300
7	GRE 321	Engineering Drawing Studio	MON	9am - 10am	B.Eng Computer Engineering	300
8	ELE 323	FL3 (Rm 008)	MON	1pm - 2pm	B.Eng Computer Engineering	300

View Files As

~Map~	Map~i	Table
<input checked="" type="radio"/> Venues	<input type="radio"/> I	<input type="radio"/> T
<input type="radio"/> Time Table	<input type="radio"/> I	<input type="radio"/> T
<input type="radio"/> Search Result	<input type="radio"/> I	<input type="radio"/> T

COURSE OF STUDY

VENUE

COURSES

COURSE ALLOCATION

ADD NEW RESOURCE

ADD NEW CATEGORY

LOGOUT

Figure 16 : Courses Allocated To Venues.

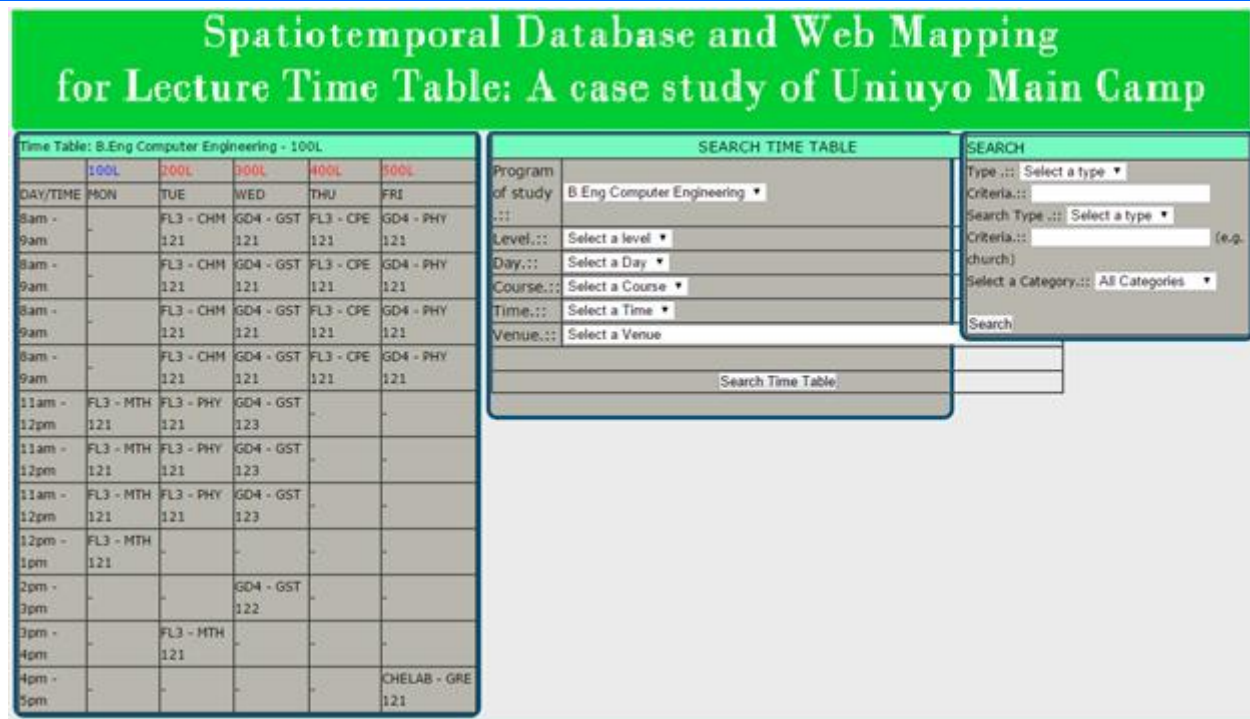


Figure 17 Search for a venue, course, level, day or time the map

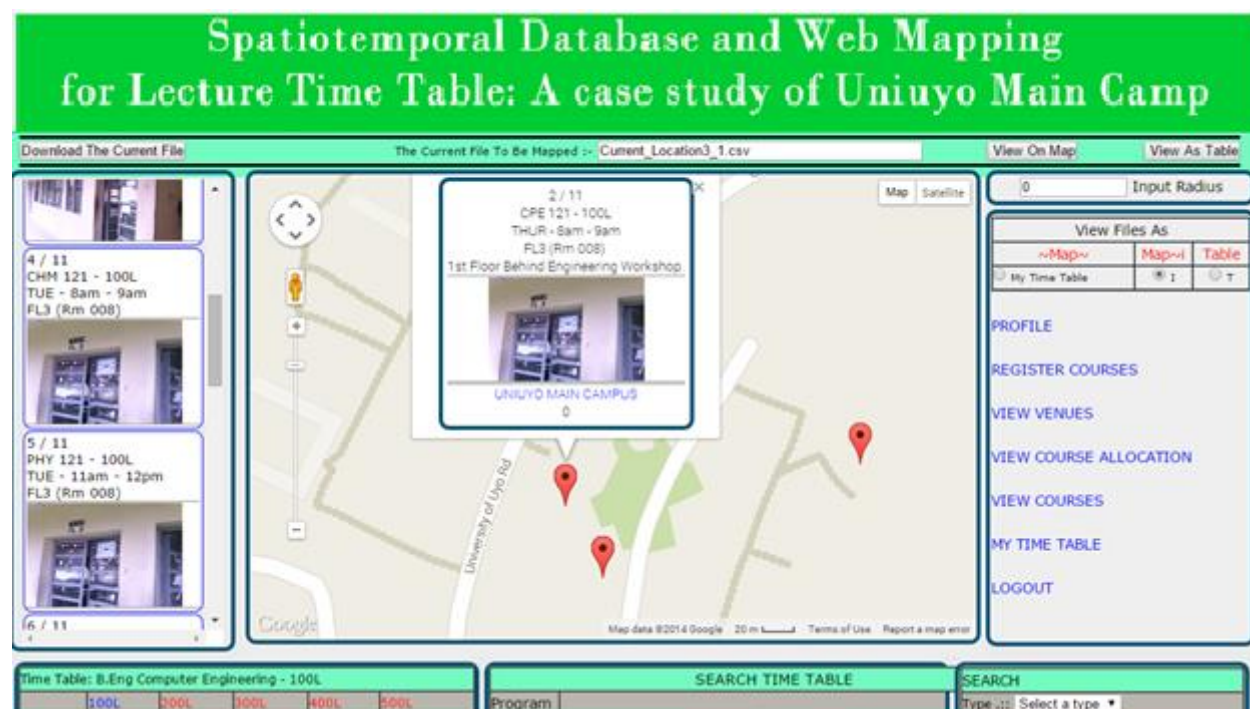


Figure 18: Search results of a lecture's venues on the map

V. CONCLUSION

The design and implementation of a web-based spatiotemporal lecture timetable system for University of Uyo is presented. The web application used Google Maps Application Programming Interface (API) to enable the visualization of the lecture venues on Google map. Waterfall software development methodology is used for the system design and implementation. The requirement engineering process

model diagram, functional decomposition diagrams, module and submodule flowcharts and usecase diagrams for the system are presented along with the table structure used in the database design. Sample lecture timetable for the faculty of Engineering University of Uyo is used to demonstrate the effectiveness of the system. With the web application, students and staff can easily locate the lecture venues using the features on Google map. They can also search and extract their personal timetable from the web-hosted school timetable.

REFERENCES

1. Hooshmand, S., Behshameh, M., & Hamidi, O. (2013). A tabu search algorithm with efficient diversification strategy for high school timetabling problem. *arXiv preprint arXiv:1309.3285*.
2. Mockus, J., & Pupeikienė, L. (2012). On multi-start algorithms for optimization of high school timetables. *Informatica*, 23(3), 405-425.
3. Mushi, A. R. (2011). Non-Linear Great Deluge algorithm for Tanzanian High Schools Timetabling. *International Journal of Advanced Research in Computer Science*, 2(4).
4. Mohammadi, M. S., & Lucas, C. (2008, September). Cooperative co-evolution for school timetabling problem. In *Cybernetic Intelligent Systems, 2008. CIS 2008. 7th IEEE International Conference on* (pp. 1-7). IEEE.
5. Imhonopi, D., Urim, U. M., Onwumah, A., & Kasumu, T. O. (2017). An appraisal of information and communication technologies as new media tools for language teaching and learning in tertiary institutions in Nigeria. *IFE Psychologia: An International Journal*, 25(1), 185-209.
6. Johnson, L., Becker, S. A., Cummins, M., Estrada, V., Freeman, A., & Hall, C. (2016). *NMC horizon report: 2016 higher education edition* (pp. 1-50). The New Media Consortium.
7. Pucciarelli, F., & Kaplan, A. (2016). Competition and strategy in higher education: Managing complexity and uncertainty. *Business Horizons*, 59(3), 311-320.
8. Edewor, P., Imhonopi, D., & Urim, U. M. (2014). ICTs and sustainable development of higher education in Nigeria: Rewriting the ugly narrative. *Journal of educational and social research*, 4(1), 357-363.
9. Imhonopi, D., & Urim, U. M. (2011). Maximising ICT for quality research and teaching in tertiary institutions in Nigeria. *Journal of Society and Development*, 1(3), 44-58.
10. Njumanze, Q. U. (2010). ICT in participatory development of teaching/learning English as a global language in Nigeria: A discourse. *African Journal of Teacher Education*, 1(1).
11. Ogunsola, L. A. (2005). Information and communication technologies and the effects of globalization: twenty-first century "digital slavery" for developing countries--myth or reality. *Electronic Journal of Academic and Special Librarianship*, 6(1-2), 1-10.
12. Walker, L., Cross, M., & Barnett, T. (2018). Mapping the interprofessional education landscape for students on rural clinical placements: an integrative literature review. *Rural and remote health*, 18(2), 1-18.
13. Kaiyan, H., & Qin, W. (2018). The development and popularisation of network platform of college sports venues in intelligent manufacturing. *International Journal of Reasoning-based Intelligent Systems*, 10(1), 20-25.
14. Campus, W. (2017). Campus Map. *Acute Care*, 1, 5.
15. Ibrahim, O. A., & Mohsen, K. J. (2014). Design and implementation an online location based services using Google maps for android mobile. *International Journal of Computer Networks and Communications Security (CNCS)*, 2(3), 113-118.
16. Hu, S., & Dai, T. (2013). Online Map Application Development Using Google Maps API, SQL Database, and ASP .NET. *International Journal of Information and Communication Technology Research*, 3(3).
17. Hu, S. (2012). Online Map Service Using Google Maps API and Other JavaScript Libraries: An Open Source Method. In *Online Maps with APIs and WebServices* (pp. 265-278). Springer, Berlin, Heidelberg.
18. Bildirici, I. O., & Ulugtekin, N. N. (2010, November). Web mapping with Google maps mashups: overlaying geodata. In *A Special Joint Symposium of ISPRS Technical Commission IV & AutoCarto in Conjunction with ASPRS/CaGIS* (pp. 15-19).
19. Peng, X., & Wu, X. (2010). Digital campus map publishing based on Google Map API. *J Geomatics*, 35(1), 25-27.