**INTERACTIVE SUPPORT SYSTEM**

**AIM AND OBJECTIVES OF THE SYSTEM**

The aim and objectives of this system is to design an online system that will enable students to lodge their support services to their lecturers without actually going to the lecturers’ offices. Other objectives of this system include:

1. To design a system that contains all the lecturers and students of the department and their profiles.
2. To enable the department to keep track of support services required by the students and support provided by the lecturers.
3. To design a system that will decongest the number of students trooping to lecturers’ offices to lodge their required support services.

**TECHNICAL TOOLS**

The system will be designed using software development tools and are classified into Front – End tools, Back – End tools and Web Server.

**Front–End Development Tools**

The front – end here means the web pages that user will be navigating through. Several front – end designing tools are available such as HTML, CSS, PHP and JAVASCRIPT.

**Back-End Development Tools**

The backend in software engineering is the part of the software that handles databases. MYSQL is selected to be used as a database development tool for the new system.

**THE WEB SERVER**

Apache web server is the best choice for its security, reliability and availability.

**SOFTWARE REQUIREMENT SPECIFICATION**

Software requirements are the set of programs modules needed to control and coordinate the activities of the hardware device of the computer system. These requirements are specifically selected to suit the programming task of the system.

Operating System – Android and Windows 7, 8 or 10

Processor Speed – 2.0Ghtz

Internet Browser – Chrome, Firefox, Mozilla or Opera

Web pages Development Environment – Notepad++

Database Development Environment – Wampserver

**4.1 SYSTEM DESIGN**

The proposed system being an online interactive system that involves students and lecturers featuring the following prospect:

* Students can send support services to lecturers
* Lecturers will receive and respond to support services sent by students
* Admin will receive and respond to support services sent by the students
* Admin is responsible for adding new staff or student and has the privileges to view all the submitted and resolved support services

**4.1.1 PROPOSED SYSTEM MODULES**

The proposed system will have three different units (modules) namely: the Admin’s unit, the lecturer’s unit and the student’s unit.

1. **The Admin’s Unit:** This is where the administrator (person controlling the site, preferably the HOD) creates/adds lecturers’ and students’ profile.
2. **The lecturer’s Unit:** This is where a lecturer view his/her profile and respond to the support services that was sent to him by students.
3. **The Student’s Unit:** This is where a student view and edit his/her profile, send support services about missing marks, and view the response about his/her support services.

**4.2 SYSTEM DEVELOPMENT METHODOLOGY**

Although there are several methods of developing software such as waterfall model, spiral model, iterative model, to mention but few, Incremental/Iterative Model of software development methodology is preferred here to be used to develop the new system. This is due to the fact that the proposed system is categorized into modules and the modules/units will be developed one after another, after which they will be integrated together to form a single one as the new system.

**4.2.1 DATA DESCRIPTION**

Generally, the new system data will be formed from three set of people, that is, from the students, lecturers, and admin. The data from the students’ side consists of the students details (profile) such as Matriculation number, surname, first name, department, level, etc. On the admin side, the data will be the admin details and finally the lecturers’ profile consists of the details such as surname, first name, position, address, marital status, etc.

Consequently, these data will be stored in five (5) different tables which include:

1. Users table: This will hold the admin, students and staff usernames and password details
2. Staff table: This will contain all the details of the staff
3. Student table: This will hold all details of the students
4. Support service table: This will hold all details about the support services
5. Resolve support services table: This will hold all details about the resolve support services

The detail structure of these tables/entities is in the subsequent part (Database Structure).

**4.2.2 INPUT SPECIFICATION**

The inputs specification for the new system can be categorized module -wise. Basically all the information is managed by the system and in order to access the information one has to produce his/her identity by entering the user-name and password. Every user has his/her method of access beyond which the access is dynamically refrained rather denied.

The input specification from the student site is the details about his/her support services which includes privileges to send support services to lecturers and to receive reply of his/her needed support services. This is in addition to his/her a profile detail which includes also matriculation number, department, gender, level, profile picture, etc.

In the admin site, the input specifications are the detail about the admin profile, new staff or students added and the details about sent and received support services.

The lecturer site has only one input specification that is response comment to the student support services.

**4.2.3 OUTPUT SPECIFICATION**

Basically, major outputs of all the modules/units in the system are reports about the students' support services. The reports, as it is obvious, carries the general picture of the whole information that flows across the system, which are created dynamically to meet the requirements on demand. However, the system when design must be able to produce output at different module for different inputs.

**4.2.4 DATABASE STRUCTURE**

As it is planned in the Data Description part of this report that the new system comprised of five (5) tables, the following is the sketch of how the entities will be designed in the database, each of which will be holding a number of fields.

Table 4.1: **‘Users’** table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field** | **Data Type** | **Length** | **Null** | **Comment** |
| Username | Varchar | 30 | No | Primary key |
| Password | Varchar | 30 | No |  |
| Role | Varchar | 20 | No |  |

Table 4.2: **‘Staff’** table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field** | **Data Type** | **Length** | **Null** | **Comment** |
| StaffID | Varchar | 45 | No | Primary key |
| Title | Varchar | 30 | No |  |
| Surname | Varchar | 45 | No |  |
| Firstname | Varchar | 45 | No |  |
| Position | Varchar | 45 | No |  |
| Phone | Varchar | 45 | No |  |
| Department | Text | 45 | No |  |
| Address | Varchar | 45 | No |  |
| Email | Varchar | 45 | No |  |
| Sex | Varchar | 45 | No |  |

Table 4.3: **‘Student’** table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field** | **Data Type** | **Length** | **Null** | **Comment** |
| MatNum | Varchar | 30 | No | Primary key |
| Surname | Varchar | 30 | No |  |
| Firstname | Varchar | 30 | No |  |
| Sex | Varchar | 30 | No |  |
| Phone | Varchar | 30 | No |  |
| State | Varchar | 30 | No |  |
| L.G.A | Text | 30 | No |  |
| Address | Varchar | 30 | No |  |
| Email | Varchar | 45 | No |  |
| Level | Varchar | 20 | No |  |
| Department | Varchar | 30 | No |  |

Table 4.2.6.4: **‘**Support Service**’** table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field** | **Data Type** | **Length** | **Null** | **Comment** |
| SupportID | Int | 11 | No | Primary key, auto increment |
| SupportBy | Varchar | 30 | No |  |
| SupportTo | Varchar | 30 | No |  |
| Comment | Text | 35 | No |  |
| SupportDate | Date |  | No |  |
| StudSupportStatus | Varchar | 10 | No |  |
| StaffSupportStatus | Varchar | 10 | No |  |
| SupportStatus | Varchar | 10 | No |  |
| ResolveDate | Date |  | No |  |

Table 4.2.6.5: **‘ResolveSupportService’** table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field** | **Data Type** | **Length** | **Null** | **Comment** |
| ID | Int | 11 | No | Primary key, auto increment |
| ResolveBy | Varchar | 45 | No |  |
| Comment | Text |  | No |  |
| DateResolve | DateTime |  | No |  |
| SupportBy | Varchar | 45 | No |  |
| SupportID | Int | 11 | No |  |
| Status | Varchar | 10 | No |  |

**4.2.7 SYSTEM FLOWCHART**

A flowchart is aimed at presenting the visual flow of data through a system, the operation performed within the system and the sequence in which they are performed. Consequently, below is how the data will be flown and the operation that will be perform in the new system when design.

Lecturer

Yes

Yes

No

View your

Support?

Log Out?

Logging out

View your Feedback?

Yes

Display your Feedback

G

Yes

E

Edit your

Profile?

Yes

Display Edit Profile page

D

Lodge support?

Yes

Display Support page

F

Display sent Support services

Yes

E

Enter

Login Details

Is login detail valid?

Contact Us

Login

About Us

Contact Us Page

About Us Page

Display Home pPaPPage

**Start**

Student Home Page

No

Admin

C

Student

Which User?

B

View your profile?

Display Student Profile page

Logging Out

Yes

A

J

B

Lecturer Home Page

View your

profile?

Edit your

profile?

View students support service?

Display lecturer profile page

Log Out?

Logging Out

A

Display Edit

Profile page

Display students

Support services

C

Add new Staff?

Display Profile

**M**

Admin Home Page

Edit your Profile?

Add new Student?

View your Profile?

View all support services?

Log Out?

Logging Out

A

Edit Profile

**N**

Add Student

**O**

Add Staff

**P**

View support ser

**Q**

**4.3 SYSTEM IMPLEMENTATION**

To start the implementation phase of the new system after having being done with the fact finding and planning phase, Notepad++ and Wamp Server are to be used during the implementation as for the creation of the front end and back end respectively.

To begin the implementation, the Notepad++ and the Wamp Server has to be installed on the system to be used. After which the database along with its tables are created. The site definition and the connection should also come after the database. Having done the above mentioned steps, the admin unit is the first to be design, then the students unit, and finally the lecturers unit. This is in accordance with the methodology chosen as the development technique.

However, the system may undergo some changes as time goes on due to observations that may come up from the observers or any other staff from the department, or some changes in the requirements specifications.

**4.4 SYSTEM TESTING AND EVALUATION**

System testing in software engineering is a Black Box testing in which the system developed is tested in order to evaluate the system’s compliance with the specified user requirements. In view of this, the user requirements of this system are categorized into three units/modules as earlier mentioned. These are the Admin requirements, the Lecturers requirements and the Students requirements.

Table 4.2.1: System Testing

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1. ***Admin Requirements Testing*** | | | | |
| **Reference** | **Requirement** | **Pass/Fail** | **Comment** | **Fixed?** |
| 1.1 | The system shall allow the admin to view his profile | Pass |  |  |
| 1.2 | The system shall allow the admin to edit his profile | pass |  |  |
| 1.3 | The system shall allow the admin to add new staff | Pass |  |  |
| 1.4 | The system shall allow the admin to add new student | Pass |  |  |
| 1.5 | The system shall allow the admin to view all support services | Pass |  |  |
| 1. ***Staff Requirements Testing*** | | | | |
| 2.1 | The system shall allow the staff to view their profiles | Pass |  |  |
| 2.2 | The system shall allow the staff to edit their profiles | Pass |  |  |
| 2.3 | The system shall allow the staff to view and respond to support services sent to them | Pass |  |  |
| 1. ***Student Requirements Testing*** | | | | |
| 3.1 | The system shall allow the students to view their profiles | Pass |  |  |
| 3.2 | The system shall allow the students to edit their profiles | Pass |  |  |
| 3.3 | The system shall allow the students to view the status of their support services | Pass |  |  |
| 3.4 | The system shall allow the student to view the feedback of his/her support services | pass |  |  |