2.2.4. Design Constraint

**Programming language:**

The programming language used is HTML, CSS, and javascript including their libraries and the programming language used was PHP. SQL queries could also be used to obtain data stored in the database.

**Operating system:**

We were asked by clients to make the services available across all platforms, along with IOS and Android.

**Framework:**

The client has not specified, so any framework we think is more appropriate can be used.

**Front-end graphic style:**

The client seeks a bright hue to make it look more enchanting and there should be the use of bold letters.

**Design approach:**

The client advised object-oriented design, which includes both top-down and bottom-up approaches.

**Software Architecture:**

The architecture will be local to a university, but there will be secondary storage sites to safely store data and for backups.

**Hardware requirements:**

The suggested system is supposed to work on any internet-connected device that has an internet browser.

**Development standards:**

The funding is confined, and a functional system must be established within it. Because it interacts with highly personal data, the system must be able to avoid System vulnerabilities.

**2.2.5. Commercial Constraints (Total Project)**

**Schedule:**

The client has set a deadline of mid-July for the design and study to be completed and the final product should be delivered to the customer by November 20th.

**Estimated Budget:**

Around £28,960

The expected budget is shown in the table below based on factors like:

1. The amount of time it took to finish the work and the estimated staffing costs.
2. The cost and the resources of the project.
3. Necessary gain margin

###### *Table 2.2.5.1 – table showing all expenditure during software development.*

|  |  |  |  |
| --- | --- | --- | --- |
| Project details |  |  |  |
| Total project duration(week): | 5 weeks |  |  |
| Earn per hour: | £12 |  |  |
| numbers of employees: | 6 |  |  |
|  |  |  |  |
| Project outline |  |  |  |
| Segments | Weeks | Weekly hours per person | Total earn |
| Requirement specification | 6 | 4 | £2,231 |
| Design and analysis | 7 | 5 | £2,456 |
| Deconstruction of project | 12 |  |  |
| Console | 28% | 5 | £2,395 |
| Website | 27% | 5 | £2,821 |
| Mobile application | 12% | 5 | £920 |
| Testing | 40% | 5 | £2,398 |
| Evaluation | 5% | 5 | £399 |
|  |  |  |  |
| Added expenses |  |  |  |
| Reason | Cost |  |  |
| Software/hardware | £6,000 |  |  |
| Office expenses (£110 per week) | £2,000 |  |  |
|  |  |  |  |
| Overall project cost | £21,250 |  |  |
|  |  |  |  |
| Other charges (10%) | £3,000 |  |  |
| Profitability (20%) | £5,500 |  |  |
| Program right | Acceptable |  |  |
|  |  |  |  |
| Overall project expense | £29,750 |  |  |

3. **System Analysis and Design (Records Management System)**

System analysis and design relate to the procedure of analyzing a company problem to optimize it through better practices and approaches. It can also be termed as the procedure of planning a new system, or one to replace or complement an existing system which basically focuses on continuous improvement and achieving profitable growth targets.

This section includes the system analysis and design documentation. The system will be built using the PHP MVC design, which separates user views, access to data, and logical web application features. All the data will be stored in a MySQL database.

3.1. Preliminary design stages

The preliminary design stage is a decision-making process whose goal is to explore if the existing system's issue or shortcoming is actual. Certain features of a project's viability could be reinvestigated by the development team. The desire to continue or terminate the contract is the eventual consequence.

3.1.1. Textual analysis

A textual test is a means of understanding how individuals make logical sense of and convey life and life events by deciphering the words, signs, and/or visuals included in writings. Graphic, textual, and spoken information often provide indications for the study demonstrated. Sociocultural systems are generally regarded as altering and reflecting ideas.

Appropriate classes and behaviors would be featured in the textual analysis contingent on the instruction, materials, and workshops that were supplied.

|  |  |  |
| --- | --- | --- |
| S.N | Applicant category | Applicant features |
| 1 | Student | Create\_student, Amend\_student, Archive\_student, Display\_student, Assign\_student |
| 2 | staff | Create\_staff, Amend\_staff, Archive\_staff, Display\_staff, Assign\_staff |
| 3 | Course | Create\_structure\_course, Amend\_course, Display\_course, Delete\_course, Archive\_course |
| 4 | Module management | Create\_module\_management, Amend\_module management, Delete\_module\_management, Archive\_ module\_management, Display\_module\_management, Assign\_modulemanagement |
| 5 | Assignment management | Create\_assignment\_management,  Amend\_assignment\_management, Delete\_assignment\_management, Archive\_assignment\_management, Display\_assignment\_management, Assign\_assignment\_management, Mark/Grade\_assignment\_management |
| 6 | Attendance records | Create\_attendance, Amend\_attendance, Archive\_attendance, Monitor\_attendance, Display\_attendance, Action\_Poor\_Attendance |

3.1.2. Significant event analysis

Significant event analysis outline some of the possible actions that various kinds of users may do when using the system.

**STUDENT**

|  |  |  |  |
| --- | --- | --- | --- |
| S.N | Events | Performer | Associated fields |
| 1 | View student | Admin | University id, full name, birthdate, gender |
| 2 | Add student | Admin | Full name, email, password, gender, birthdate, contact, address |
| 3 | Edit student | Admin | Full name, email, university id, birthdate |
| 4 | Delete student | Admin | University id, full name, password, gender, birthdate, contact, address |

**STAFF**

|  |  |  |  |
| --- | --- | --- | --- |
| S.N | Events | Performer | Associated fields |
| 1 | View staff | Admin | Id, full name, duty, address |
| 2 | Add staff | Admin | Full name, email, password, duty, address, gender, birthdate |
| 3 | Edit staff | Admin | Full name, email, Id, duty, birthdate |
| 4 | Delete staff | Admin | Id, full name, gender, contact, address |

**COURSE**

|  |  |  |  |
| --- | --- | --- | --- |
| S.N | Events | Performer | Associated fields |
| 1 | View course | Admin, student, module\_leaders | University id, full name, birthdate, gender, password, address, email, address, status, role |
| 2 | Add course | Admin, module\_leaders | University id, full name, gender, address, password, contact, email, role, status |
| 3 | Edit course | Admin | University id, full name, birthdate, password, email |
| 4 | Delete course | Admin | University id, full name,  gender, password, birthdate, email, address, contact, role, status |

**MODULE MANAGEMENT**

|  |  |  |  |
| --- | --- | --- | --- |
| S.N | Events | Performer | Associated fields |
| 1 | View module | Admin  Student  Module\_leaders | University id, full name, gender, password, address, birthdate, email, contact, status role |
| 2 | Add module | Admin | University id, full name, gender, password, address, birthdate, email, contact, status role |
| 3 | Edit module | Admin | University id, full name, gender, password, address, birthdate, email, contact, status role |
| 4 | Delete module | Admin | University id, full name, gender, password, address, birthdate, email, contact, status role |

**ASSIGNMENT MANAGEMENT**

|  |  |  |  |
| --- | --- | --- | --- |
| S.N | Events | Performer | Associated fields |
| 1 | View assignment | Module\_leaders  Students | Student Id, role, status, course id |
| 2 | Add assignment | module\_leaders  students | Student Id, role, status, course id |
| 3 | Edit assignment | Admin | Student id, course id, role, status |
| 4 | Delete assignment | Admin | Course id, status |

**ATTENDANCE RECORDS**

|  |  |  |  |
| --- | --- | --- | --- |
| S.N | Events | Performer | Associated fields |
| 1 | View attendance | Admin  Module\_leaders | University id, full name, password, gender, birthdate, role, contact, email, address, status |
| 2 | Add attendance | module\_leaders | University id, full name, status, |
| 3 | Edit attendance | Admin | Student id, full name, email, status |
| 4 | Delete attendance | Admin | Student id, full name, password, email, address, status |

**3.1.3 Commands queries and constraints**

The guidelines necessary to interact with a database to perform tasks, operations, and analyses of data are classified commands. One can make use of commands to access data as well as perform multiple activities such as designing tables, inserting data into tables, updating data, and removing tables. Queries are an information appeal to the database. Constraints are guidelines for how data in a table should be dealt with.

**STUDENT**

|  |  |  |  |
| --- | --- | --- | --- |
| CLASS | *STUDENT* | | Part: 1/1 |
| TYPE OF OBJECT  Append, view, notify, delete and upgrade students | | Author: Bipana  Created: 05/12/2022 | |
| Queries | view\_lecture, view\_assignment, view\_attendance, view\_grade, view\_ announcement, view\_module, append\_files, download\_lectures, upload\_assignment | | |
| Commands | First name, middle name, surname, course, level, address, birth date, email, contact, submit | | |
| Constraints | * Admin adds, updates, and deletes students * students can only view the modules in which they are enrolled * Students can access lectures and materials by downloading them * All students enrolled in that course are notified * Students can submit assignments online * has access to grades | | |

**STAFF**

|  |  |  |  |
| --- | --- | --- | --- |
| CLASS | *STAFF* | | Part: 1/1 |
| TYPE OF OBJECT  Append, view, delete, notify, upgrade staff | | Author: Bipana  Created: 05/12/2022 | |
| Queries | Add\_staff, view\_staff, delete\_staff, edit\_staff, update\_staff | | |
| Commands | Staff id, full name, email, gender, address, role, contact, birthdate | | |
| Constraints | * Admin has the ability to append, delete and view staff * Staff generally looks after the courses * The staff gets notified when enrolled * They are divided with their roleplay | | |

**COURSE**

|  |  |  |  |
| --- | --- | --- | --- |
| CLASS | *COURSE* | | Part: 1/1 |
| TYPE OF OBJECT  Append, view, notify, delete and upgrade courses | | Author: Bipana  Created: 05/12/2022 | |
| Queries | add\_course, delete\_course, update\_course, edit\_course, assign\_module, add\_lecture, download\_lecture, | | |
| Commands | Course topic, course illustration, course leader, submit | | |
| Constraints | * The module leader has the ability to add, upgrade, and remove courses. * Students could only visualize lectures from the module in which they are enlisted. * Students can download lectures. * A notice is being sent to all students enrolled in that course. | | |

**MODULE MANAGEMENT**

|  |  |  |  |
| --- | --- | --- | --- |
| CLASS | *MODULE* | | Part: 1/1 |
| TYPE OF OBJECT  Append, view, delete and upgrade module | | Author: Bipana  Created: 05/12/2022 | |
| Queries | University id, Full name, Password, Gender, Birthdate, Address, Contact, Email, Role, Status | | |
| Commands | Append, Alter, login, Logout, Change Password, Archive, Delete | | |
| Constraints | * Administrator is essential to look after the module * Manager should be able to manage all the courses accordingly * Courses should be frequently updated | | |

**ASSIGNMENT MANAGEMENT**

|  |  |  |  |
| --- | --- | --- | --- |
| CLASS | *ASSIGNMENT* | | Part: 1/1 |
| TYPE OF OBJECT  Append, view, notify and download assignments | | Author: Bipana  Created: 05/12/2022 | |
| Queries | Add\_assignment, upload\_assignment, submit\_assignment, download\_assignment, delete\_assignment, assign\_assignment, notify\_grade | | |
| Commands | Append, View, Download, Archive, Delete | | |
| Constraints | * Assignments could be created, modified, deleted, and notified. * Assignments are available for download. * Assignments include a grade. | | |

**ATTENDANCE RECORDS**

|  |  |  |  |
| --- | --- | --- | --- |
| CLASS | *ATTENDANCE RECORDS* | | Part: 1/1 |
| TYPE OF OBJECT  Append, view, notify, delete, and upgrade attendance | | Author: Bipana  Created: 05/12/2022 | |
| Queries | add\_attendence, view\_attendence, update\_attendence, notify\_attendence, delete\_attendence | | |
| Commands | Append, Alter, mail, Delete | | |
| Constraints | * The module leader has the ability to add, upgrade, and remove attendance. * Students can only look at their attendance records. * A notification will be sent to all students enrolled in that module. | | |

3.3 Detailed dynamic system design

3.3.1 Events chart

3.3.2 Object creation chart

3.3.3 System scenario chart

3.3.4 Dynamic diagrams

3.4 system database design

Database design is a series of procedures that aid in the creation, advancement, execution, and upkeep of organization data processing. The project cannot proceed unless a database is designed. A well-designed database is simple to sustain and enhances data reliability whose primary goals are to generate physical and logical concepts of a suggested database system to satisfy the needs of users and perform admirably.

3.4.1 E-R model

After determining all database tables, ERD was created to ensure that the comprehensive relation between software tables was noticed. The concluded ERD is shown below, and it shows a complete linkage between entities.

Diagram

Description automatically generated

3.4.2 Attribute listings

After developing the database structure and deciding on the connection attribute linking the databases, the database attributes were concluded and are displayed in the table below.

The database entities are as outlined.

The following table details the attributes for every table.