|  |  |
| --- | --- |
|  |  |
| rgu logo colour |
|  | |

|  |  |
| --- | --- |
|  |  |

|  |  |
| --- | --- |
| **COURSEWORK**  **DUE DATE** | 14/12/2019 |
| **MATRIC No** | **1909247** |
| **SURNAME** | **Chatzileftheris** |
| **FIRST NAME(S)** | **Stefanos** |
| **COURSE TITLE** | **IT for Oil and Gas Industry** |
| **MODULE NUMBER**  **& TITLE** | **CMM007 – Intranet System Development Part 3** |
| **LECTURER ISSUING COURSEWORK** | **Stewart Massie** |

**RGU Research Paper Project (CMM007-PART 3)**

**Abstract:**

In this report I am going to analyse with all the necessary details my project. I will recommend some new techniques that I could add into my project and I will explain what kind of system architecture I am using and how can I improve that. Moreover, I will give some scalability solution and give some idea for the security.

Description: This project was designed to implement a web application that will help group project to upload, identification, review, monitoring and following of research papers between the group members. The research activities will be report writing, review and other similar tasks can be to monitor and track the editorial progress of several papers within a limited time.

1. **Analysis of current limitations:**

The name of this project is RGU research paper, in this system they will be three types of users: Administrator, Student Team Leader and Students. All users will have access to the register system i.e. members will be required to log in. The administrator has the highest role to manage the tools on the platform, and he will only have the access to create users, setup project groups, allocate team leader role and assign members to the project. The team leader is responsible for delivering papers to members for review and. Students will be able to upload research papers, this will be submitted to the Student team leader, who will allocate it to any members for review. Furthermore, this project will give the opportunity for the user to create platform that will support group project teams.

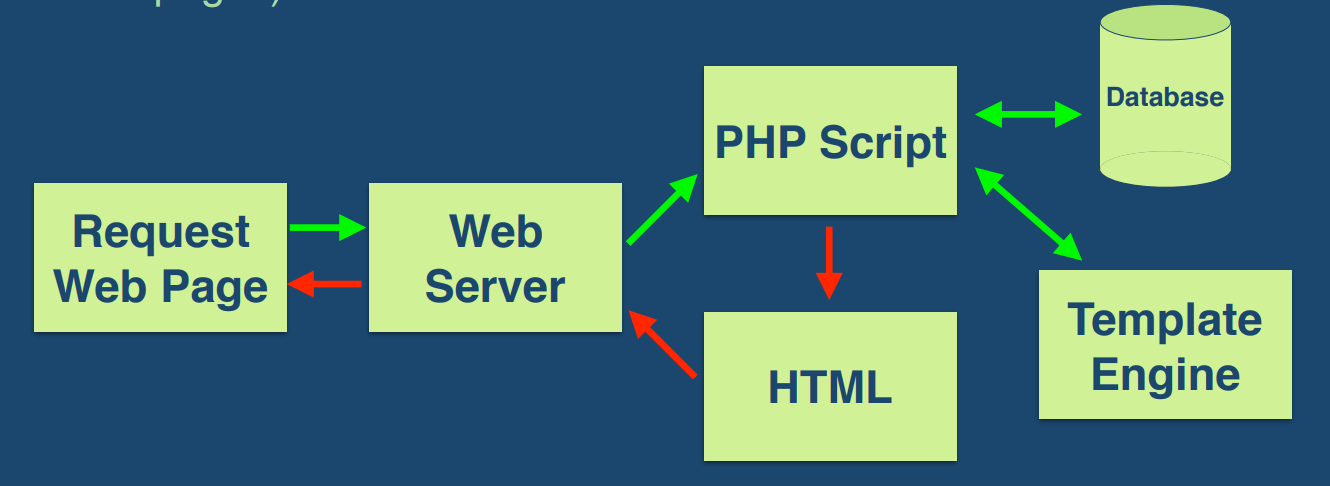
The created web application will contain the following features. More than one user role, the application will be able to support users with different roles assigned various degrees of permissions and benefits in what a user is able of doing. Some type of file upload system and: This will facilitate the uploading of research papers on the server for storing and download at request. This will be implemented using HTML and PHP for executing upload to remote server. A system for user to input data that is stored and then recalled from a database: The system will support data storage for example documents or reviewing papers and retrieval using HTML, PHP in the application and MySQL(phpMyAdmin) for the database server. The application will be hosted on a local server (Xampp) accessible only with a request.

In addition, the application must contain both front end (client side) and (server side) code. The front-end is the presentation layer that the user will use to perform tasks on the web app. The front-end is the user-friendly Graphical User Interface built. It is built using HTML, CSS, Font awesome and Bootstrap. The backend is the logic and data layer. It executes user's requests, by performing query based on the request and return response to the user on the front-end.

1. **Recommended Technologies:**

An alternative way to design my database will be much better than the

***Server-side technologies:***

******The application code is usually stored on the server. Customers make requests to the servers. Then the server responds to these requests after gathering the required information. I can use mobfire us a database LAMP)

***Client-side technologies:***

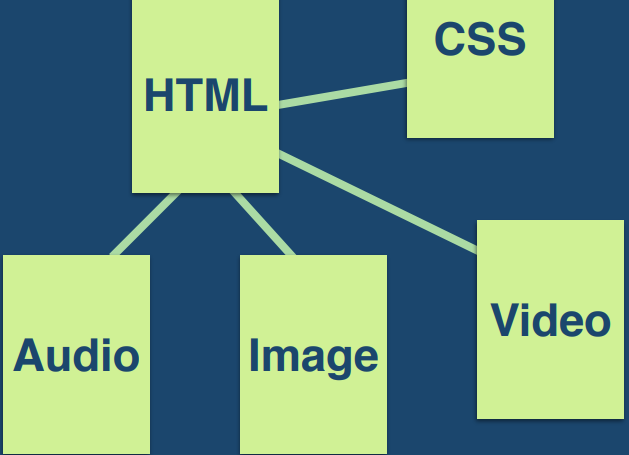
Each user of a client application called. Clients can be computers, mobile, tablet, etc. Usually, many customers interact with the same application that is stored on the server.

Look and feel improved with style sheets

presentation and style could be separated

from the information structure and content

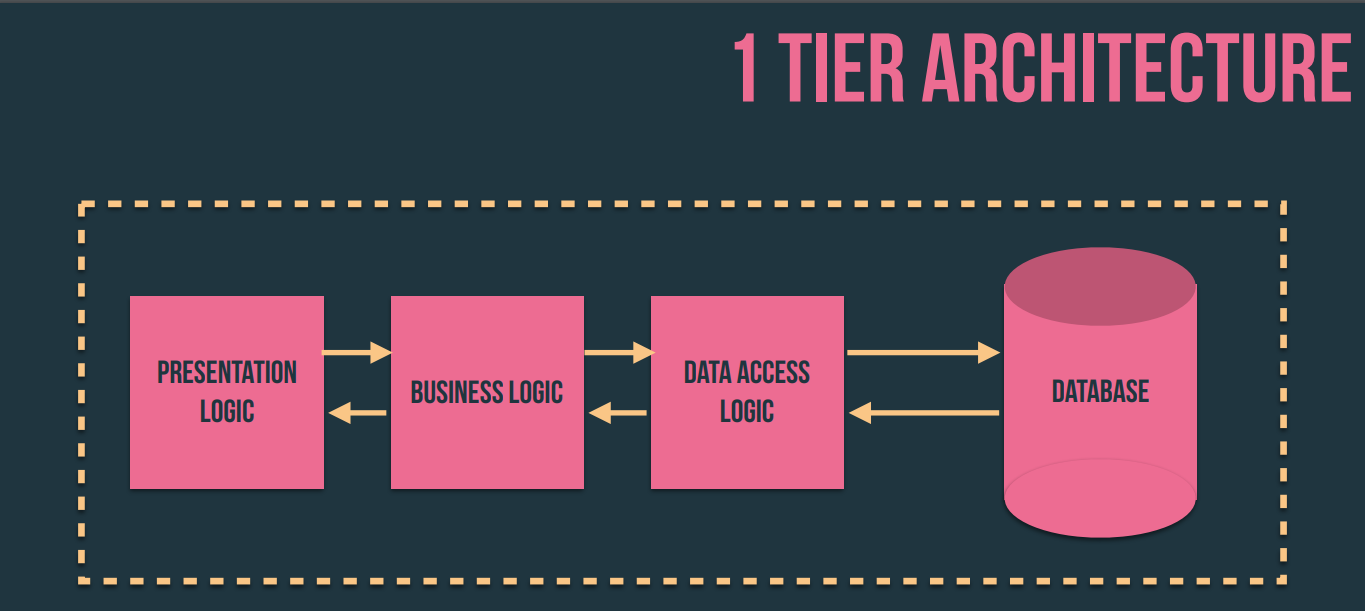
Cascading Style Sheet (CSS)

Mixed Media

1. **Proposed system architecture:**

***3.1) What I am using***

I am using the first-tier architecture for my project because of the situation. I only use a localhost from my apartment xampp to do all the necessary changes in php for my project. My project follows the three layers down below which are presentation, business and data layer, which I am going to explain the layer down below.



**Presentation Logic Layer:** Using this level, we can access the websites. The primary function of this layer is to communicate with the application layer. This level transmits the information supplied by the user in terms of keyboard events, mouse-click the application level. In a simple word, it is to see the implementation.

**Application/Business Logic Layer:** It is also known as the Business Logic Layer, which is also known as Layer Logic. According to the example of the login page with the message, after the user clicks the Connect button, the application layer interacts with the level of the database and sends the required information to the presentation level. It checks the functionality of an application by making detailed processing. This layer acts as an intermediary between the level of presentation and database.

**Data Logic Layer:** The data stored at this level. The level of application communicates with the database level to recover the data. It contains methods that link the database and perform the required actions, such as: insert, update, delete, etc. Which in this situation I am using xampp (localhost).

On the other hand, the first tier is very bad for scalability (single processor being used), bad for portability and bad for maintenance (if you want to change one thing).

***3.2) Improvements for system architecture***

In my opinion the best architecture system is the N-tier because every layer can be on a different machine, easier to maintain, components are reusable, and it will have faster division of work. So, there are several benefits to using n-tier architecture for your software.

**Secure:** You can secure each of the three tiers separately using different methods.

**Easy to manage:** You can manage each tier separately, adding or modifying each tier without affecting the other tiers.

**Scalable:** If you need to add more resources, you can do it per tier, without affecting the other tiers.

**Flexible:** Apart from isolated scalability, you can also expand each tier in any manner that your requirements dictate.

In short, the architecture n-tier, you can adopt new technologies and add more information without having to rewrite the entire application, or redesigning your entire software, making it easier to scale and maintain. Meanwhile, in terms of security, you can store sensitive or confidential information in logical level, keeping them away from the presentation level, thus making it safer. Other benefits include: More efficient development, Easy to add new features and Easy to reuse.

1. **Scalability Solutions:**
2. **Application Security:**
   1. A login system: This is a security mechanism to restrict application access to authorized users. This will be created using HTML, PHP, MySQL CSS and Bootstrap.

**Conclusion:**

**References:**

<https://www.softwaretestingmaterial.com/software-architecture/>

<https://stackify.com/n-tier-architecture/>

lecture 1

lecture 7

lecture 9

lecture 10