CS303: Software Engineering

**Project Deliverable 3: System Decomposition, Architecture, and Updated Plan**

# Team Composition

Monisa Hassan Alvi (17L-4098)

Aiza Nadeem (17L-4338)

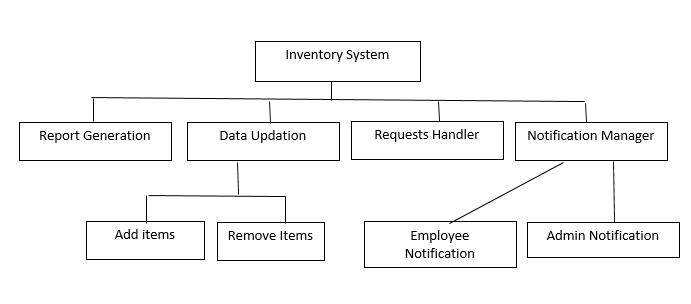
Bisma Jalal (17L-4557)

Tehreem Mushtaq (17L-4325)

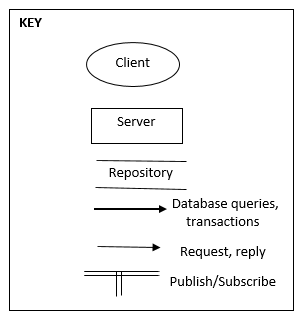
M. Abdullah Farooqi (17L-4122)

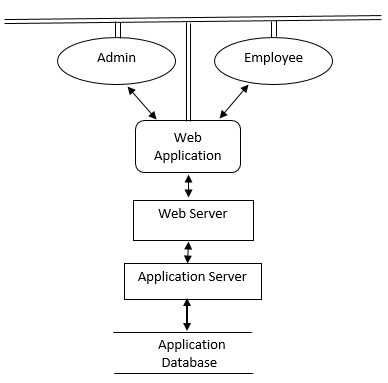
**Part 1:**

**Functional Decomposition:**



**Architecture:**





**Architectural Styles:**

The architectural styles which we are going to follow in this project are:

* Client-Server
* Repository
* Publish and Subscribe
* Layering

**Client-Server**

The client which is either an employee or admin in this case will send a request to the server. All employees can send requests to the server for acquisition of items. Server will respond that the employee's request got approved by the admin or not. The admin will send a request to the server to login the system. The server will allow the admin to login if correct credentials are entered. After this, the admin can perform other tasks by requesting the server. In this way client server architecture is being followed.

**Repository**

The Inventory Management System is a real-time database managing inventory items. This system will keep track of items issued to employees in its database. It will have a specific threshold for every item stored in its database and will notify the administrator when some item is below a certain threshold. It will allow the admin to add new items, issue items, delete items to the existing inventory database.

**Publish and Subscribe**

The system also follows publish and subscribe architecture. The admin needs to be notified for different events and employees also need to be notified if some item is overdue or his item acquisition request is approved or not. In this way, the system will follow publish and subscribe architecture.

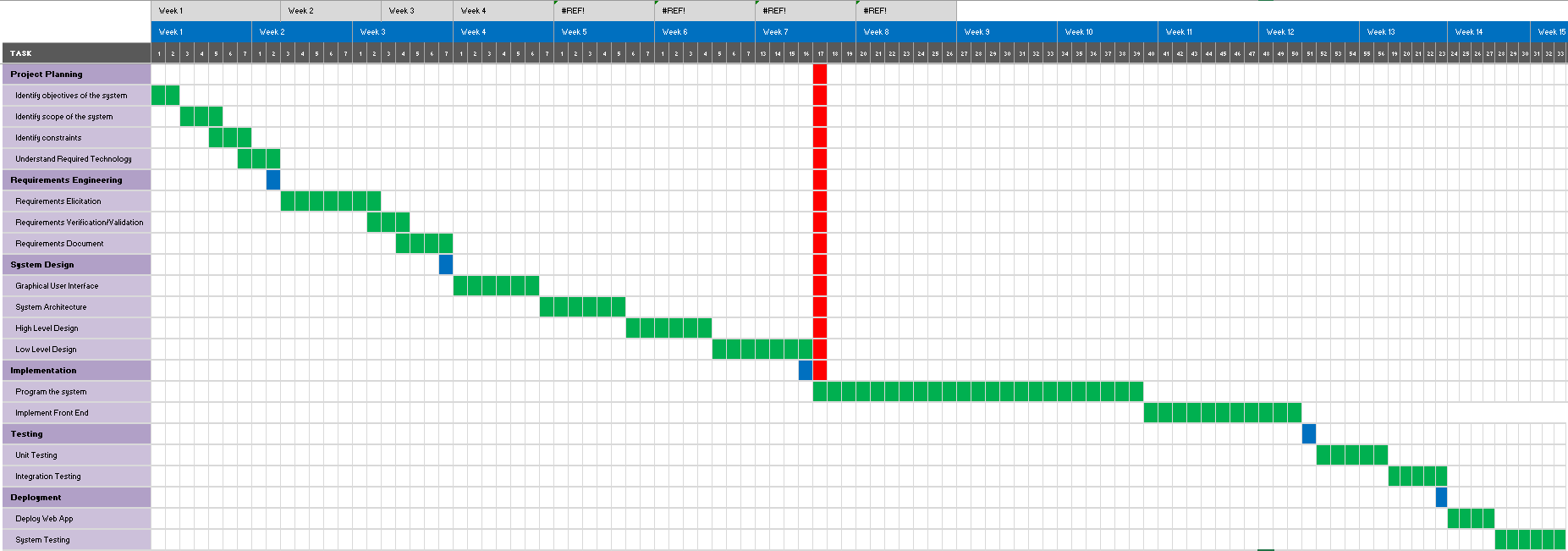
**Layering**

Layering will also be employed by the system. The users will interact with the web application using the browser. The web application will then pass the client (users) request to the web server. The web server will then interact with the application server. The application will interact with the repository. In this way, abstraction is guaranteed by using layering architecture.

**Part 2:**

|  |  |  |  |
| --- | --- | --- | --- |
| Task | Assigned To | Performed By | Remarks |
| Project Planning  (identify objectives, scope, constraints and required technology for the system) | All members | All members | Done by conducting 3 team meetings |
| Context Diagram | Aiza, Tehreem | Aiza, Tehreem |  |
| Graphical User Interface | Bisma | Bisma |  |
| Use Cases | Abdullah, Monisa | Abdullah, Monisa |  |
| Requirements Gathering (elicitation, verification, documentation) | All members | All members | Includes requirements meeting with Sir Zeeshan and Sir Taimur. Includes 2 group meetings. |
| AON Graph | Aiza | Aiza |  |
| Gantt Chart | Bisma, Tehreem | Bisma, Tehreem |  |
| Functional Decomposition | Abdullah, Aiza | Abdullah, Aiza |  |
| Architectural Strategies Identification | Aiza, Monisa, Abdullah | Aiza, Monisa, Abdullah |  |

**Updated Gantt Chart:**



The red line specifies the work that has been done till now.

\*Please zoom to read tasks from the Gantt chart.