Software Architecture used For University Boighar

Noakhali Science and Technology University

Institute of Information Technology (IIT)

Bachelor of Science in Software Engineering (BSSE)

Prepared BY

Foyzul Karim Pathan

ID: ASH1825033M

1. Introduction OF University Boighar

The main purpose of this project is to provide the automation service of buying books, sells unused books, easily can access academic all books and various other features. Mainly we want to build an online book store. They want to build an app which will connect five parties including book buyer, book seller, IT club, freelancer and system admin. They all will be connected through a large database which will help them to store their data and get the necessary information without any hazard. We also visualize their connection through different diagrams. This project will reduce time and cost of both parties and it will be effective for our education sector alsoIn the current situation of our country, it is very hard for buying our academic books from Dhaka. All academic books are not found on library in everywhere. Every time it's not possible to buy books in a collective way. It causes havoc for all students. We live in the era where we can get data through access and used internet, we can connect any person in everywhere. But all time it's not possible to academic books buying manually. All academic books are not found in the university library. It wastes time, cost and create various other problems. As no such system exists, where we can be buying our academic books easily and selling unused books and earn money via internet. The ultimate result costs and tome of the students are waste. They can't access the books in their needed time. By analyzing such circumstances, we want to build an automation system of proper online book store system through our project "University Boighar". All information of buyer, seller, freelancer and It club member will be stored and they can easily access their data very easily. We also ensure the necessary security of information and other processes.

2. User Classes and characteristics

There are six types of stakeholders in our "University Boighar". Such as:

Book Buyer: Book buyer can view the available book on the system and also request for his/her needed books. Buyer can get the books in short time and minimum cost. But first of all, buyer need to create an account to access this feature. They can view the book list very easily.

Book Seller: Book seller can easily sell his/her unused books easily. To sell unused book can also earn money. On the other side, another user can easily buy seller unused books with minimum cost and short time.

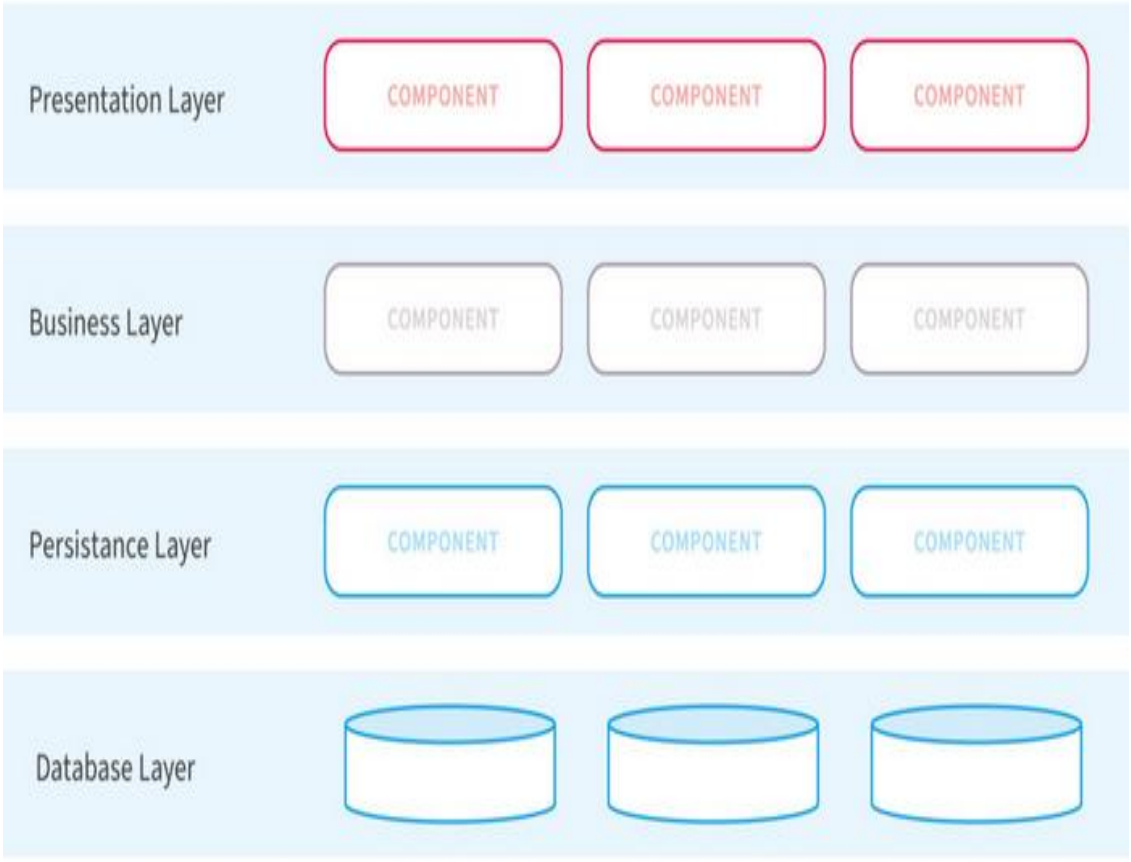
Admin: Admin manage the web site such that manage book list, user request, user information, IT club member information, system freelancer information.

Freelancer: freelancer will have an account. He/She can view the book request and accept all valid request and easily can earn from the system.

3. Architecture Process and Patterns for University Boighar

- **Layered Pattern**

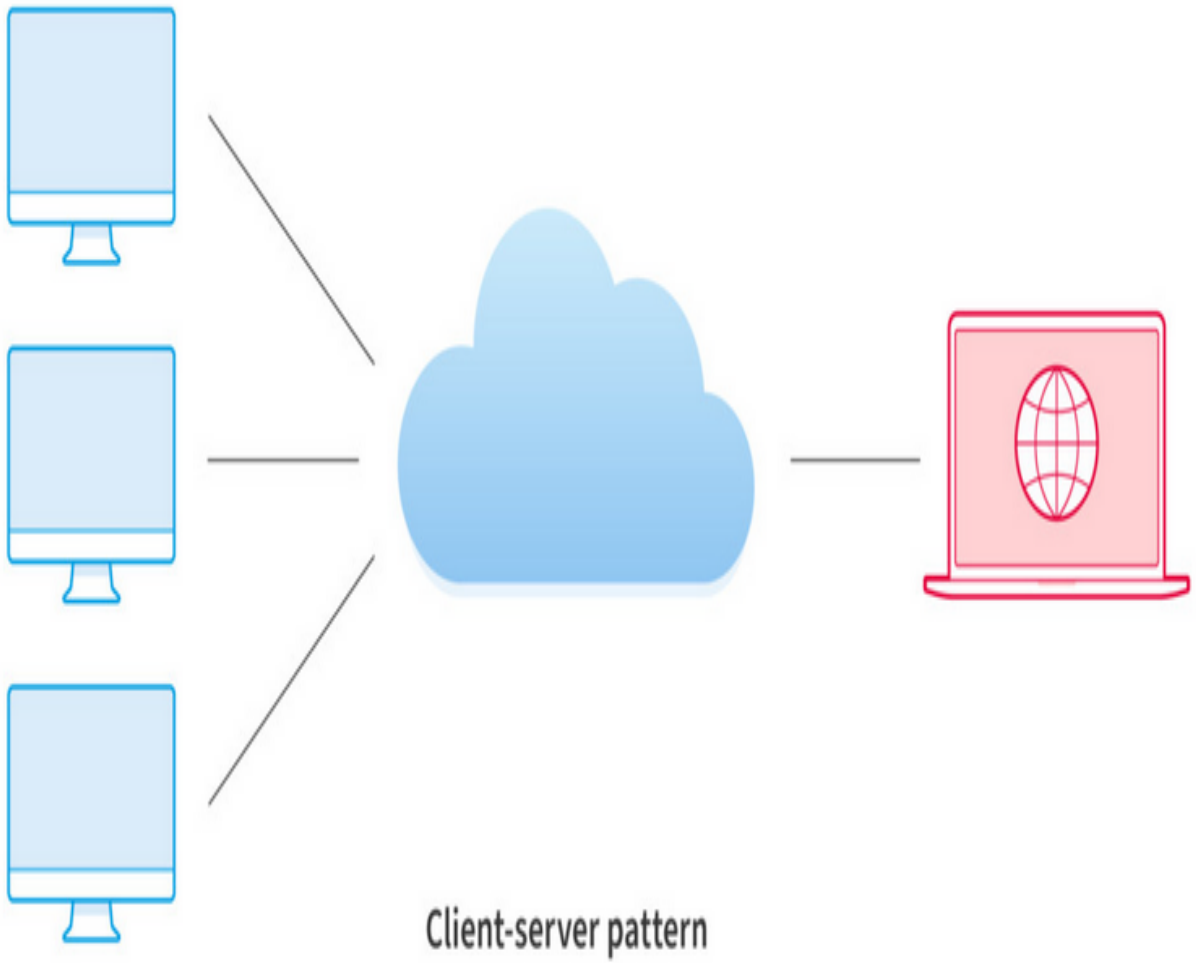
University Boighar follows Layered Pattern because in this pattern are separated into layers of subtasks and they are arranged one above another. Each layer has unique tasks to do and all the layers are independent of one another. Since each layer is independent, one can modify the code inside a layer without affecting others.



Layered pattern

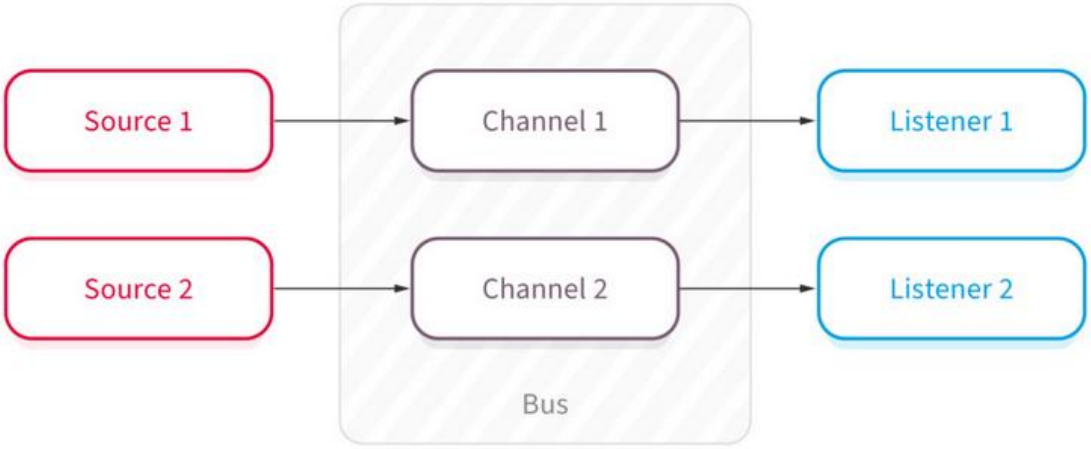
- ## Client-Server Pattern

The client-server pattern has two major entities. They are a server and multiple clients. University Boighar follows client server architectural pattern. Client sends request method to the server and based on this request server will be response.



- ## Event-Bus Pattern

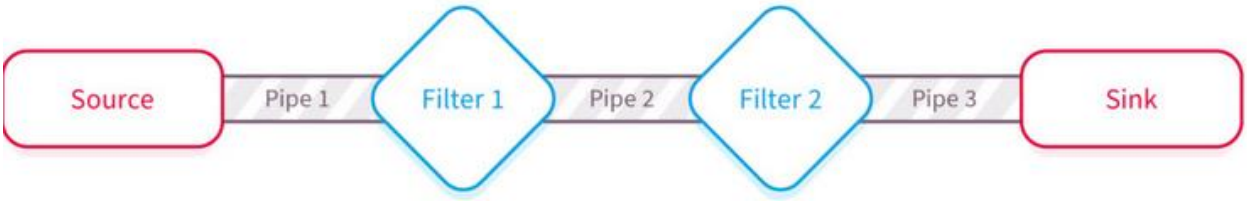
Event-Driven Architecture is an agile approach in which services (operations) of the software are triggered by events. When a user takes action in the application built using the EDA approach, a state change happens and a reaction is generated that is called an event. A new user fills the signup form and clicks the signup button on Boighar and then a Boighar account is created for him, which is an event.



Event-bus pattern

- ## Pipe-Filter Pattern

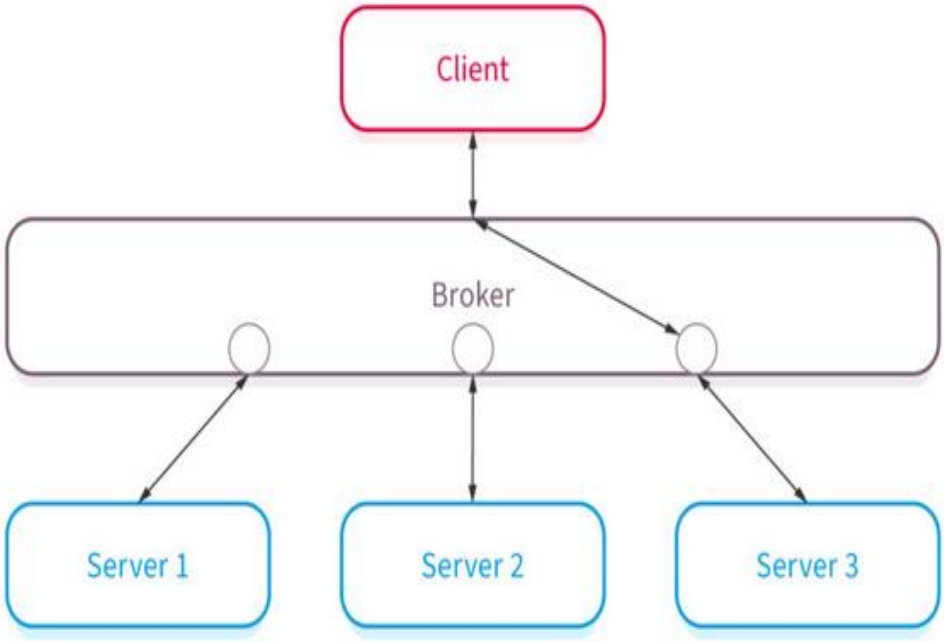
Boighar have complex processing in hand. we will likely break it down into separate tasks and process them separately. This is where the “Pipe-filter” architecture pattern comes into use. There are repetitive steps such as reading the source code, parsing, generating code, etc. These can be easily organized as separate filters. Each filter can perform its’ processing in parallel if the data input is arranged as streams using pipes.



Pipe-filter pattern

- **Broker Pattern**

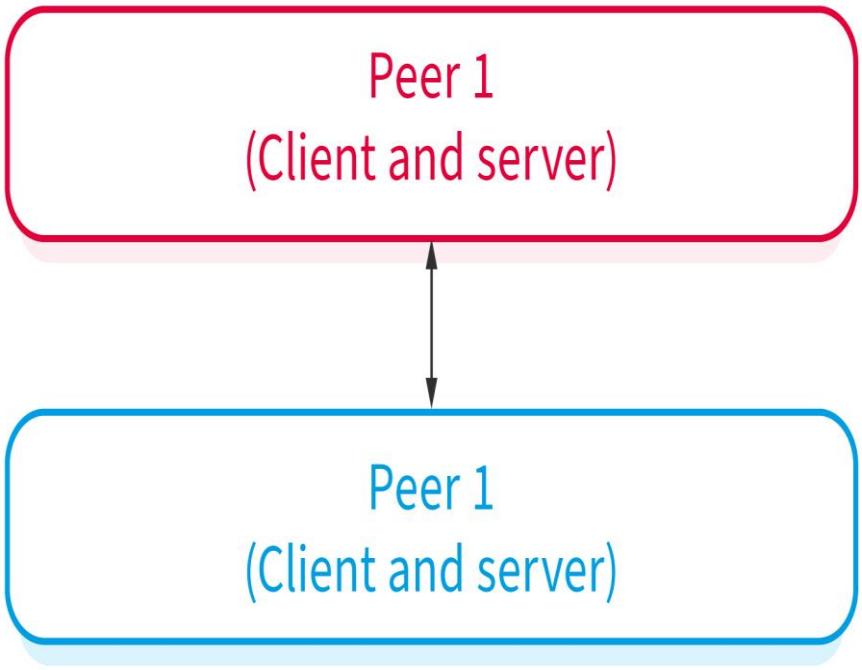
Consider distributed systems with components that provide different services independent of each other. Independent components could be heterogeneous systems on different servers, however, clients still need their requests serviced. “Broker architecture pattern” is a solution to this. University Baighar follows that pattern.



Broker pattern

- **Peer-to-peer Pattern**

Peer-to-peer pattern is markedly different from the client-server pattern since each computer on the network has the same authority. P2P networks are decentralized, therefore, they are more secure. You must have already heard a lot about the security of the Bitcoin network.



Peer-to-peer pattern

4. Architecture Analysis

1) Usability

It is described as how the user is utilizing a system effectively and the ease of which users can learn to operate or control the system. The well-known principle of usability is KISS. Software applications should be user-friendly. University Boighar is a user friendly system.

2) Reliability

It is the ability of a system to continue to keep operating over time. Reliability has sometimes been classified as "how quality changes over time." The difference between quality and reliability is that quality shows how well an object performs its proper function, while reliability shows how well this object maintains its original level of quality over time, through various conditions.

3) Availability

It is the ratio of the available system time to the total working time it is required or expected to function.

4) Portability

It is the ability of a software application to run on numerous platforms such as data portability, hosting, viewing, etc.,

5) Testability

It shows how well the system or component facilitates to perform tests to determine whether the predefined test criteria have been met.

6) Scalability

It is the ability of a system to handle the demand for stress caused by increased usage without decreasing performance.

7) Flexibility

It is the ability of a system to adapt to future changes

9) Maintainability

It is the ability of a software application to maintain easily and support changes cost-effectively.

10) Supportability

It is the ability of a system that satisfies necessary requirements and needs to identifying and solving problems.

11) Interoperability

It is the ability of two or more systems to communicate or exchange data easily and to use the data that has been exchanged.

12) Performance

It is the ability of a system in the form of responsiveness to various actions within a certain period of time

13) Security

It is the ability of a system to resist or block malicious or unauthorized attempts that destroy the system and at the same time provide access to legitimate users.

5. conclusion

we need to take the next step and adopt the perspective that a software architecture is, fundamentally, a composition of architectural design decisions. These design decisions should be represented as first-class entities in the software architecture and it should, at least before system deployment, be possible to add, remove and change architectural design decisions against limited effort.