

Architectural Study of Software Engineering, Advantages and Challenges

Palika Jajoo

Assistant Professor in Computer Science,
SKIT Jaipur
palika@skit.ac.in

Abstract: - In order to create software application, it is necessary to understand the concept of the software engineering. This paper will describe what is software engineering and how it is followed to deliver a software application. In this paper the architecture of the software will be discussed. There are many categories of architecture of software and patterns. For each category of architecture, there are few advantages as well as challenges in the implementation of the software.

Keywords: - Software engineering, Architect of Software, Challenges, Advantages. Categories of software architecture.

Introduction: - The concept of software engineering defines the process of designing and manufacturing the software. It is the process which follows the identification of the specifications, designing, coding, testing and implementation of the software to meet the users need. The types of engineers involved in each stage will be different. In simple words the process of converting user specifications in to working and functional software is the concept of software engineering. Whereas, the architect of the software explains the structure of the software, the components involved in the software, explains the interaction of the software with user etc. It will also define the designing of the software, the standards and protocols it will follow.

Architectural Design of Software: - [1]

In order to start creating architect of the software, the architect need to follow certain steps in order to meet users demands. Following are the different stages of the architect design of software life cycle: -

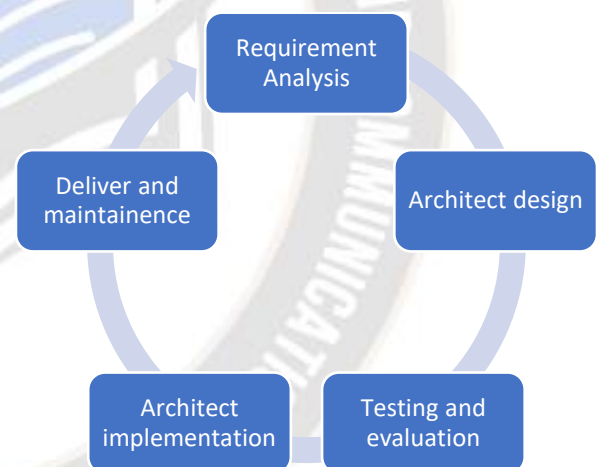


Figure 1 Architect Design Life Cycle.

- **Architect Requirement Analysis:** - In the first stage like any other life cycle, the architect will gather all the information regarding the software to be implemented. He will study the user's specifications in details and will prepare his own architect requirement document which will have all the necessary details which will be needed in future stages. Based upon this document the architect will analyse the software in detail and check for any unrealistic demand of the user.
- **Architect Design:** - Based on the document of the previous stage, in this step the actual architect designing of the software starts. The architect will decide the components of the software, how they will interact with

each other and platform on which the software will work, the protocols and standards which are mandatory to be followed. The detailed explanation of the same will be given using graphs, flow charts etc.

- Testing and evaluation: - Once the architect designer gives the design in the previous stage, it is tested whether it is according to the user's specifications. For this the architect design of the software is given to the user and once the user is satisfied then it is proceeded further otherwise again the requirement document is updated and the whole process is repeated till the time the design meets all of the user's demands. If the user wants to modify some of the specifications, then it can be done at this stage.
- Implementation and maintenance: - Once the user is satisfied with the provided architect solutions and designs, then the design is given to the other team which helps to convert the architect design of the software into actual functional software. If the coder finds any difficulty in understanding the architect design, then he can take help from the architect as he will provide full support to implement the architect design in actual working software.

Advantages of Software architect: - [2]

- It is very important to design architect of the software as it is structured way of creating a software application.
- It helps in delivering efficient functional software to the user.
- The coder can build software application accurate as the architect design of the software helps the coder to understand the internal functioning of the software easily.
- It helps to provide the delivery of the software much faster.
- The risk involved in the manufacturing of the software reduces as it will have every detail of each component of the software and how it interacts with each other which gives the opportunity to the coder to implement it faster.
- Since the internal details of the software is shown clearly in the architect design so if there are any bugs then it can be fixed faster by following the design architect.

Types of Software Architecture: - [3]

There is variety of software architecture available. The architect can follow one of these patterns in order to design the architect of the software. These are reusable patterns which can be re-used any number of times by the architect. They can even be modified as per user's needs.

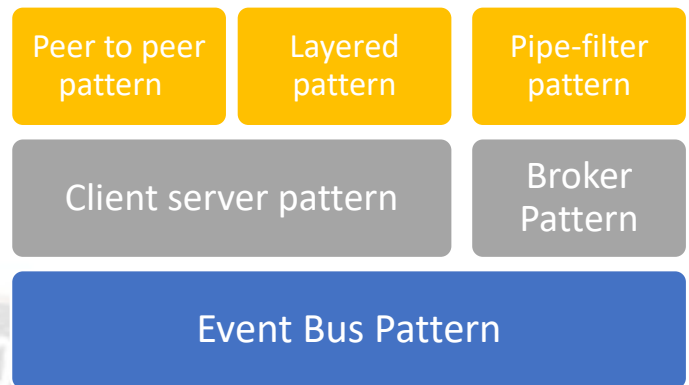


Figure 1 Architect Pattern.

1. Peer to Peer Pattern: - In this architecture pattern, there are few devices which act as client and other as server. This is also known as P2P architecture. Each component in this type of architecture has certain roles and responsibilities. The devices used in such patterns can change their roles over the period of time. Which means, a peer which acts as client can also act as server in some other query. So, this also follows client-server pattern in the same network. This can be used to share files in the same network.

Advantages of Peer-to-Peer Architect Pattern: -

- There is no need for separate clients and server to implement such kind of architect design as the same is possible using the devices present in the network.
- The cost of implementing such architect design is less as compared to the other patterns.
- No need for separate team to monitor the servers.

Disadvantages of Peer-to-Peer Pattern: -

- Since the client and server is present in the same network, it might give performance issues as all the users using at the same time can definitely slow the performance of the network.
- There might be chances of security attacks.
- Since the files and folders are saved on individual device, the sharing of files can be difficult if the owner of that particular device is not around.

2. Layered pattern: - This type of architecture has different layers of the software. Each layer performs its own subtasks which can be moved on to the next layer. Each layer performs its individual task and each layer is independent of each other.

Layers of Layered Architecture: - [4]

- Presentation Layer: - This is also called as user interface layer as it will handle all the users query and is in direct contact with the end user.
- Business Layer: - This layer will deal with the particular business request raised by the user. It will process the users request and performs to give results of the user query.
- Persistence Layer: - This layer will actually provide necessary solution to the user query raised in the previous layer. It will do calculations to give result to the user query which will be passed on to the next layer which is the presentation layer.
- Presentation Layer: - The responsibility of this layer is to provide or present the solution of the user query to the end user.

Advantages of Layered Architecture: -

- As the software is build using different layers of the architect, it is easy for the different teams to work independently on their work which improves the efficiency of the software delivered.
- The whole software is divided into different layers so the coding and testing of individual components can be done without depending upon other teams.

Disadvantages of Layered Architecture: -

>This type of architecture is useful for implementation of large and complex software's but it is not useful for small projects.

- The cost of employing members for different teams is high.

3. Pipe Filter Pattern: - This type of architect design is used for the applications which follows streaming if the data. The units of the software are connected to the filters from where they travel through the pipes in order to process the data, hence the output of one becomes the input for the other.

Advantages of Pipe-filter architecture: -

- It gives efficient results in the implementation of the operating systems.
- It is easy to understand as the data is processed in stream.
- This type of pattern can be reused in future.

Disadvantages of Pipe-filter architecture: -

- This type of pattern is not user friendly.
- If the data is processed in slow speed than the entire performance of the application will be reduced.

4. Client-server pattern: -This type pf pattern follows typical client server method of accessing and processing the data. The client and the server will be in different networks. The client from one network will raise a query which can be related to any type service demand, on the hand the server in the other network will identify the request and respond with the results.

Advantages of Client-server Architect pattern: -

- All the data and the information shared can be centralised with this technique.
- The security level of this type pf pattern is higher than the other.

Disadvantages of the Client-server Architect Pattern: -

- A dedicated team is needed to deploy such architect pattern.
- Implementation of servers is costly.

5. Broker Architect Pattern: - This type pf approach is used where the devices are scattered at different places. The broker acts as a medium to facilitate communication between the client and the server. First of all, the client will send its query to the broker and from there the broker will invoke the particular server which has the capacity to respond to that particular request.

Advantages of Broker Architect -pattern: -

- The client can raise a request from any remote location through broker.
- The different components can be integrated using this technique.

Disadvantages of Broker Architect: -

- If there are a lot of queries raised by different clients from various remote locations than the response time will be slow.
- Sometimes the performance can be very low as the whole communication takes place through broker.
- If there is issue within the broker than the whole communication process will halt till the time the broker is up and running.

6. Event Bus Pattern: - In this type of pattern, the devices are attached to common medium through which they will communicate. Whenever the request is made than it will be sent to all the devices attached to it. The server or the device which has the answer to the query raised will send the response through

the same medium. The rest of the devices can discard the same.

Advantages of Bus-Event Pattern: -

- Cost effective
- Improved performance
- Satisfied customers as they receive response soon.

Disadvantages: -

- High maintenance cost.
- If there is issue with the event bus medium through which it communicates then the whole system will crash.

Conclusion: - It is very important to implement architect design of the software as it makes easy for fast coding of the software application. It gives the facility to understand the users needs in detail based on which it becomes convenient for the coders to start coding the software. From this paper it is clear that there are variety of architect patterns available, hence the architect designer can use one of these based upon the requirements. The advantages and disadvantages of each pattern is discussed in this paper.

References: -

- 1.https://www.researchgate.net/figure/Phases-of-the-software-architecture-development-lifecycle_fig1_260250237
- 2.<https://dzone.com/articles/15-benefits-of-software-architecture>
- 3.<https://towardsdatascience.com/10-common-software-architectural-patterns-in-a-nutshell-a0b47a1e90134>.
- 4.<https://www.oreilly.com/library/view/software-architecture-patterns/9781491971437/ch01.html>