

BAHRIA UNIVERSITY (KARACHI CAMPUS)

Software Design & Architecture (SEN-221)

ASSIGNMENT # 3 – Spring 2023

Problem-Based Learning (PBL) Based on: CLO-4

Class: **BSE-4B** Submission Deadline: **10th Jun 23**

Course Instructor: **ENGR. MAJID KALEEM** Max Marks: **08**

GROUP MEMBERS:

* Mutayyab Imran (02-131212-063)
* AHSAN SAJJAD (02-131212-049)
* ADEEB-ul-HASSAN (02-131212-002)

**INSTRUCTIONS:**

You have to complete this assignment in a group; maximum of three (03) students. This assignment is based on the following PBL attributes:

1. The problem should involve **real world scenarios**.
2. The problem should require students to make **reasoned decisions** and to defend them (investigation and critical analysis).
3. If used for a **group activity**, the problem should involve collaboration and group discussions.

* Present your understanding in the class and upload your findings on LMS as an assignment in **.DOCX** format.
* Assignment will be done in a group; however, each member *must* upload it on LMS individually.

Scenario:

Online collaborative applications are software tools or platforms that enable people to work together on shared tasks or projects over the internet, regardless of their physical location. These applications typically provide a range of features and functionalities to support collaboration, such as real-time communication tools, document sharing and editing capabilities, project management tools, and version control systems.

Examples of online collaborative applications include:

1. ***Project management tools*** such as Trello, Asana, and Basecamp that allow teams to manage and track project progress, assign tasks, and collaborate on project-related. documents.
2. ***Communication tools*** such as Slack, Microsoft Teams, and Zoom that enable real-time communication and collaboration through text, voice, and video chat.
3. ***Document sharing and editing tools*** such as Google Docs, Dropbox, and OneDrive that allow multiple users to collaborate on documents simultaneously, track changes, and provide feedback.
4. ***Code collaboration tools*** such as GitHub, Bitbucket, and GitLab that enable developers to collaborate on code repositories, manage code versions, and track changes.
5. ***Online collaborative applications*** have become increasingly popular in recent years, as more and more people work remotely or in geographically distributed teams. These applications can help increase productivity, reduce communication barriers, and improve teamwork by providing a centralized platform for collaboration.

Suppose you have designed an online collaborative application. Now want to make sure that the architecture for the application you have selected is correct. For this purpose, you want to apply any of the architecture evaluation techniques.

List down the names and merits & demerits of various *(at least 5)* architecture evaluation techniques and what do you think which architecture evaluation technique is well suited for this type of software application?

**BASECAMP:**

**Basecamp** is a web-based project management and collaboration platform made to assist teams with task organization, communication, and effective teamwork. Teams may manage projects, monitor progress, share files, and interact with one another on one centralized platform.

Users can build projects and divide them into separate tasks using Basecamp. Team members can be given specific tasks with due dates and priorities to complete. It offers a timeline view to show project status and completion dates.

Basecamp's messaging system, which enables team members to connect and work in real-time, is one of its core features. To ensure successful communication, users can publish messages, comment on topics, and mention certain team members.

The names and merits & demerits of various architecture evaluation techniques are given below

**Scenario-based evaluation:**

This method uses event sequences that the system is designed to handle. This method specifies how the system behaves. These scenarios are used to evaluate how well the design handles various interfaces, data flows, and system behaviors. It identifies positives and negatives and suggests ways to advance architectural design.

**Merits:**

* It is easy to use and understand
* It may be used to evaluate a number of systems and identify potential problems early in the development cycle.

**Demerits:**

* Creating a scenario can take a long time.
* It can be difficult to list every possible result.
* Measuring the system's performance under actual load circumstances may be difficult.

**ATAM (Architecture Tradeoff Analysis Method):**

This type of architecture evaluation is more extensive and considers a wider range of factors, such as performance, security, and maintainability. The architectural trade-offs of the system are honed in and evaluated by ATAM through a series of workshops with stakeholders.

**Merits**

* the ability to evaluate a range of systems
* the ability to identify potential problems early in the development cycle
* the ability to help ensure that the system meets the needs of the stakeholders.

**Demerits:**

* It may be expensive and time-consuming to complete
* require a skilled facilitator
* be difficult to include all of the parties.

**Security evaluation:**

This kind of security assessment focuses on the system's security architecture, which is the group of security-related components and rules that protect the system from unauthorized access, use, modification, or disclosure. Among the methods that can be used to assess security are risk assessment, threat modelling, and penetration testing.

**Merits:**

* Might improve the security of the system.
* Can help to lessen the risk of security breaches by being able
* to identify potential security flaws early in the development process.

**Demerits:**

* It may be difficult to involve all the stakeholders
* it requires a professional security analyst; and it can be time-consuming and expensive to accomplish.

**Performance evaluation:**

This technique evaluates the performance of the system by measuring reaction time, throughput, and scalability. Performance can be assessed using a variety of methods, including as benchmarking, stress testing, and load testing.

**Merits:**

* Can help discover potential performance bottlenecks early in the development process, which can help reduce the risk of performance issues.
* Can help guarantee that the system satisfies the performance requirements.

**Demerits:**

* It is time- and money-consuming to accomplish
* it requires a skilled performance engineer; and it may be difficult to involve all of the stakeholders.

**Usability evaluation:**

This method evaluates the system's usability, or how simple it is for users to carry out their tasks using the system. To assess usability, a variety of approaches can be utilized, including user surveys, usability tests, and cognitive walkthroughs.

**Merits:**

* Can aid in ensuring that the system is simple to use.
* Can uncover possible usability issues early in the development phase.
* Can aid in reducing the likelihood of user frustration.

**Demerits:**

* Can be time-consuming and costly to accomplish.
* Requires a trained usability specialist.
* Can be tough to get all parties engaged.

**Question #: Which architecture evaluation technique is well suited for this type of software application?**

For evaluating the **architecture of Basecamp,** the **Architecture Tradeoff Analysis Method (ATAM)** would be a well-suited technique. The **ATAM** tool was created primarily to assess the trade-offs between various architectural design choices and their effects on quality aspects including performance, security, usability, and more.

The architecture of Basecamp, a sophisticated web-based project management and collaboration tool, is key to achieving the needed functionality and performance. By using ATAM, you may evaluate how well the architecture will be able to achieve the project's quality objectives and identify any potential hazards or trade-offs.

By using **ATAM**, you can analyze the **architecture of Basecamp** by considering various quality attributes, such as:

* **Performance**: Analyze how the architecture manages resource utilization, response times, and scalability under various loads.
* **Security**: Analyze the architecture's capacity to safeguard private user information, stop intrusions, and fix any possible security issues.
* **Usability**: Look at how the architecture enables a user-friendly design, simple navigation, and effective team collaboration tools.
* **Reliability**: Analyze the architecture's fault tolerance, disaster recovery, and availability of the system.
* **Maintainability**: Examine the architecture's adaptability, modularity, and maintenance-friendliness while taking future upgrades and improvements into account.

You may priorities architectural decisions, detect potential hazards, and comprehend the tradeoffs in different design options with ATAM. It offers a methodical way to assess the architecture, take into account the viewpoints of various stakeholders, and match the architecture to the needs and goals of the project.

**Architectural Style:** Basecamp, a project management tool, follows the **MVC (Model-View-Controller) architectural design pattern.** The Model represents the data and business logic, the View displays the user interface, and the Controller handles the interaction between the Model and View, facilitating communication and updating the user interface based on user actions. This pattern helps in separating concerns and promoting code organization and maintainability.