

TEAM ABCFG HOTEL MANAGEMENT SYSTEM

Jacqueline Chavez

Devon Alonzo

Ben Gingrich

Dustin Bates

Brandon Figueroa Ugalde

Overview

- Our goal is to program a management system.
 - This system will serve customers by allowing them to make accounts and make new reservations under their account, view past reservations, and store payment information for future use.
 - This system will allow staff and to manage all activities related to room availability and room reservations.
-

Requirements

- This system should be accessible online so that customers can make reservations from anywhere as long as they have a network connection.
 - The system must be consistent to avoid conflicts, changes must be shown in real-time.
 - The system must store a record of all active reservations, past reservations, users and room occupancy; therefore, the system must have central server side functionalities to serve this purpose.
 - The website must offer security measures as desired by the client.
 - Overall, the website must be dynamic in order to fit the requirements of the client.
-

Architectural Style Choices

MVC architecture:

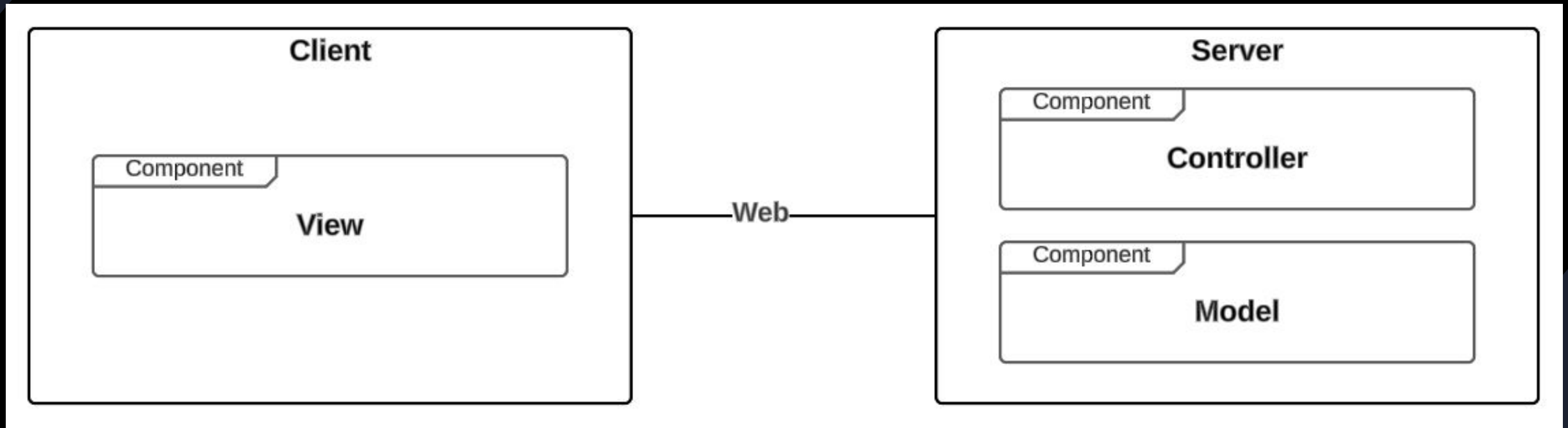
This sort of architecture offers the opportunity to organize those files that hold and process data, display data, and those that control the interaction between these two types of data. Since Jakarta EE files, the developing software we are using, are usually comprised of JSP files, html files, java files, and xml files and these all fall into one of these three categories, this architecture was amongst our top candidates.

Client-Server Architecture:

This sort of architecture is ideal for web applications such as the one we are planning to implement, therefore this architecture would serve us well. Since the client desires there to be a record of reservations, users, rooms, etc., using a centralized server integrated with a database would be ideal. This sort of architecture be efficient to implement security measures as well.

For our architecture, we will utilize a combination of both of these architectural style choices. This is because we already have files that fit the MVC architecture and we also plan to implement a database/server that will serve these files dynamically. Therefore, a combination of both is ideal.

Architecture Diagram



Conclusion

- Our system utilizes MVC architecture and the Client-Server pattern to streamline room reservations and improve customer satisfaction. However, during development, potential issues and risks include:
 - ***User Accessibility:*** Ensure accessibility for all customers.
 - ***Real-time Updates:*** Provide accurate room availability.
 - ***Self-Service Modifications:*** Allow customers to modify bookings online.
 - ***Mobile Compatibility:*** Develop a mobile-friendly platform.
 - ***Transparent Booking:*** Offer visibility into details and costs.
 - ***Stakeholder Engagement:*** Maintain alignment with project objectives.
-