

Software Design Specification

(SDS)

Report

Table of Contents

1. Introduction
 2. System Overview
 3. Use Case Diagram
 4. Activity Diagram
 5. State Diagram
 6. Component Diagram
 7. Deployment Diagram
 8. Class Diagram
 9. Conclusion
-

1. Introduction

1.1 Purpose

The purpose of this Software Design Specification (SDS) report is to provide a detailed description of the architecture, design, and functionality of the Tic Tac Toe application. This document serves as a blueprint for the development team and stakeholders, ensuring a common understanding of the system's design.

1.2 Scope

This SDS covers the design aspects of the Tic Tac Toe application, including class structure, interactions, state transitions, and deployment. The intended audience includes software developers, testers, and project managers.

2. System Overview

The Tic Tac Toe application is a desktop-based game allowing users to play Tic Tac Toe against another player or AI agent. The system includes user authentication, game configuration, and game logic components.

3. Use Case Diagram

3.1 Description

The Use Case Diagram illustrates the functional requirements of the Tic Tac Toe application. It shows the interactions between the system's users (actors) and the use cases, representing the major functions of the system.

3.2 Diagram

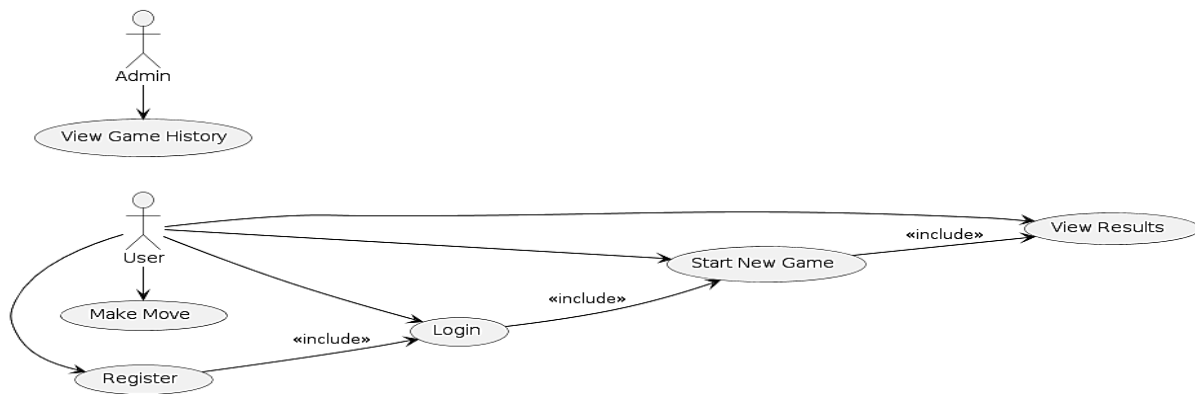


Figure 1: Case diagram

Comment: This diagram identifies the primary interactions that users (regular users and admins) will have with the Tic Tac Toe system, including logging in, registering, starting a new game, making moves, viewing results, and viewing game history.

4. Activity Diagram

4.1 Description

The Activity Diagram provides a dynamic view of the system by illustrating the flow of activities involved when a user makes a move during the game. It captures decision points and parallel activities within the process.

4.2 Diagram

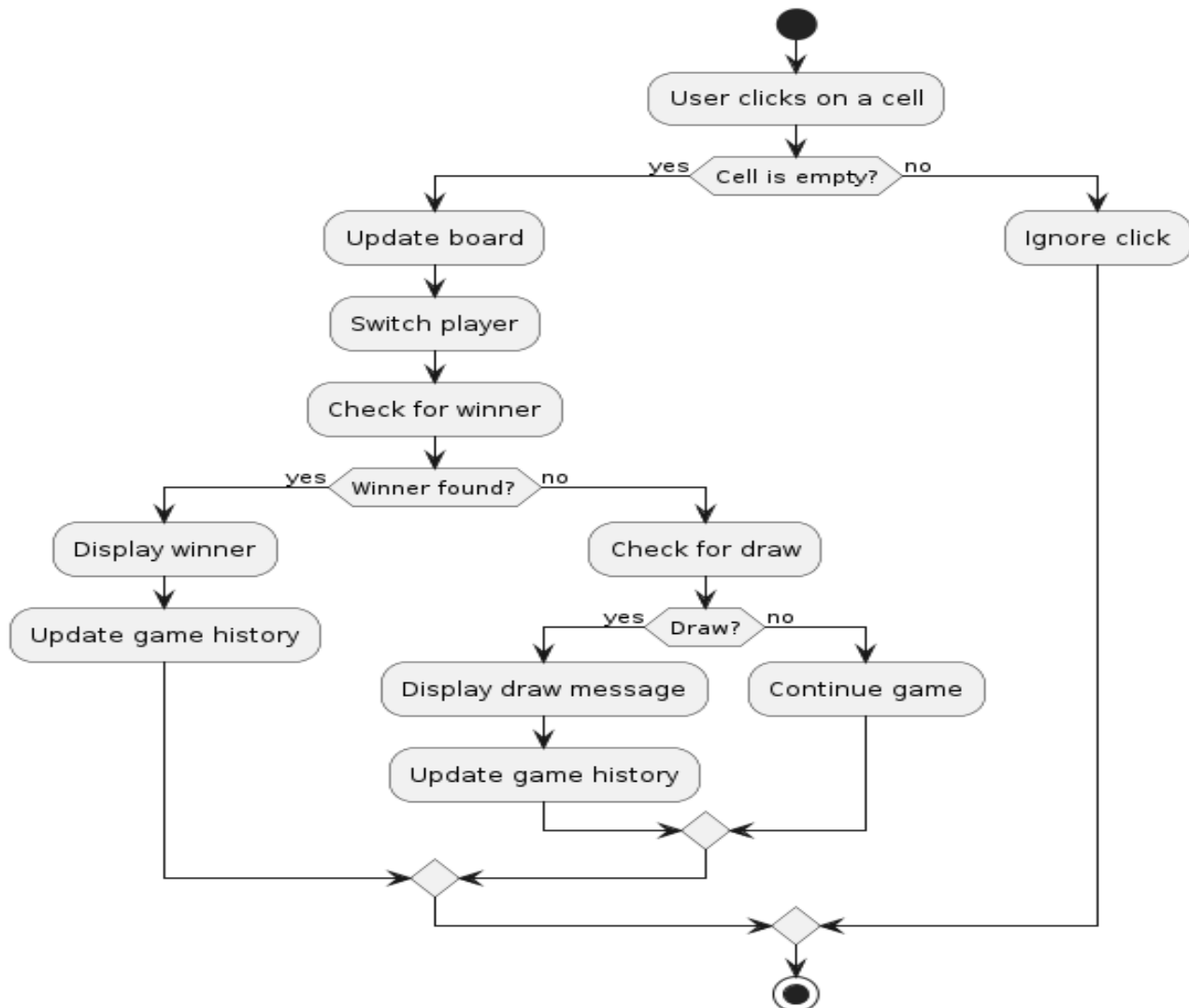


Figure 2: Diagram for making a move

Comment: This diagram details the step-by-step process that occurs when a user makes a move in the game, including checking if the move is valid, updating the game state, and determining if there is a winner or a draw.

5. Class Diagram

5.1 Description

The Class Diagram represents the static structure of the system by detailing the classes, their attributes, methods, and the relationships between them.

5.2 Diagram

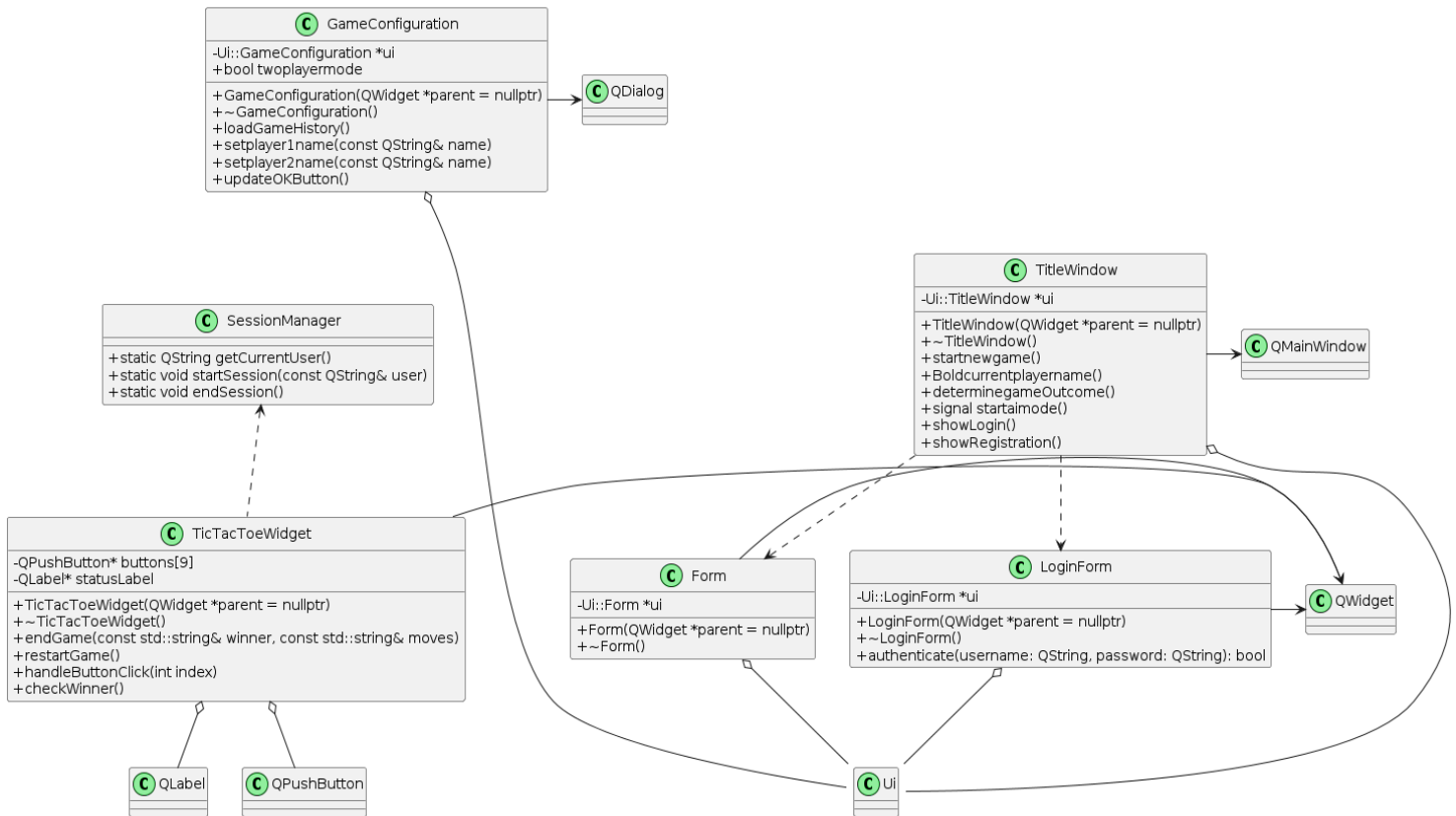


Figure 3: Class diagram

Comment: This diagram provides a detailed view of the classes within the system, including their attributes, methods, and relationships, forming the blueprint for the application's structure.

6. State Diagram

6.1 Description

The State Diagram represents the various states of the `TicTacToeWidget` and the transitions between these states based on user actions and game events.

6.2 Diagram

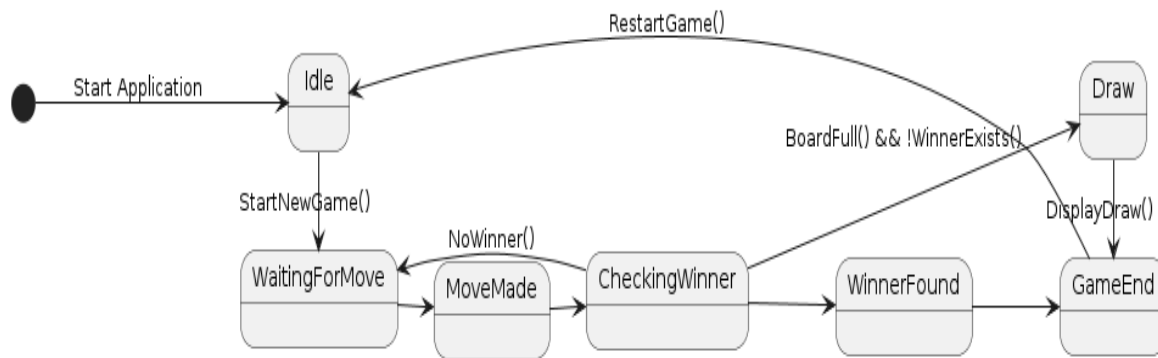


Figure 4: A simple state diagram

Comment: This diagram shows the different states of the game widget from the initial idle state to handling moves, checking for winners, and ending the game.

7. Sequence Diagram

7.1 Description

The Sequence Diagram illustrates the interactions between different objects in the system to complete a specific use case, in this case, the process of making a move during a Tic Tac Toe game. It shows the sequence of messages exchanged between the objects, detailing the flow of control and data. This diagram helps in understanding the dynamic behavior of the system by depicting the order in which operations are executed.

7.2 Diagram

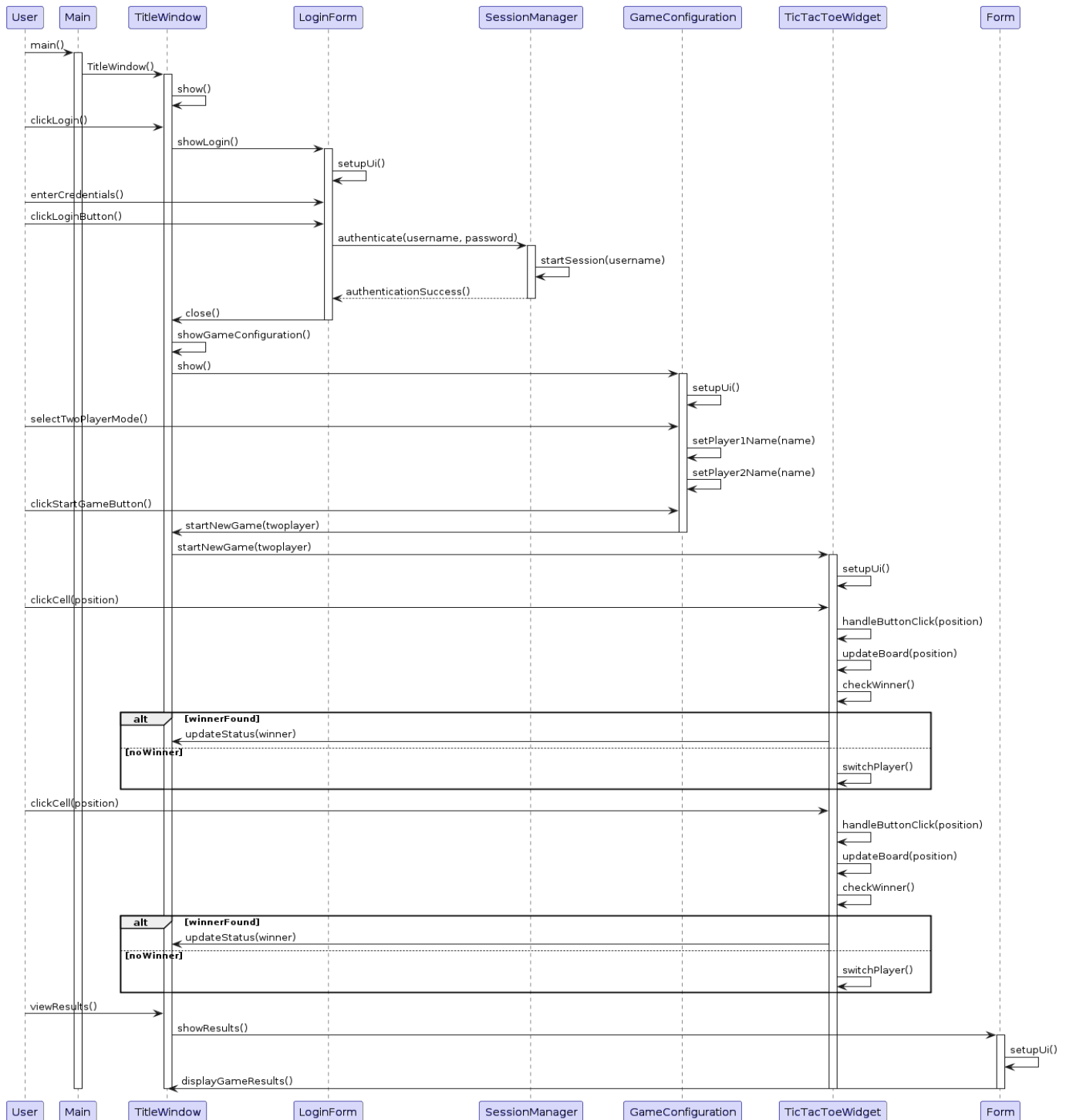


Figure 5: Sequence diagram

Comment: This sequence diagram provides a detailed view of the interactions between the user and various system components during the process of making a move in the Tic Tac Toe game. It

shows the sequence of events and the flow of messages exchanged to handle the game logic, validate moves, check for winners or draws, and update the game state accordingly.

8. Component Diagram

8.1 Description

The Component Diagram illustrates the high-level structure of the system, showing the main components and their interdependencies. This helps in understanding how different parts of the system interact with each other.

8.2 Diagram

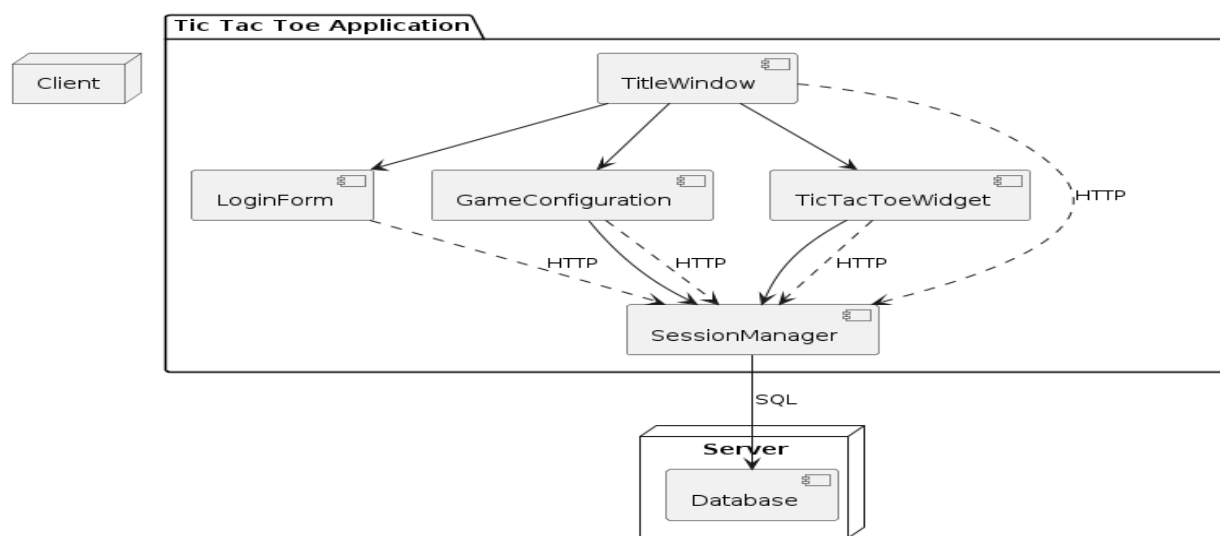


Figure 6: Component diagram

Comment: This diagram presents an overview of the system's components, showing how the client-side application interacts with the server-side session manager and database.

9. Deployment Diagram

9.1 Description

The Deployment Diagram shows the physical deployment of the software components on various nodes (hardware), illustrating the system's hardware topology and software distribution.

9.2 Diagram

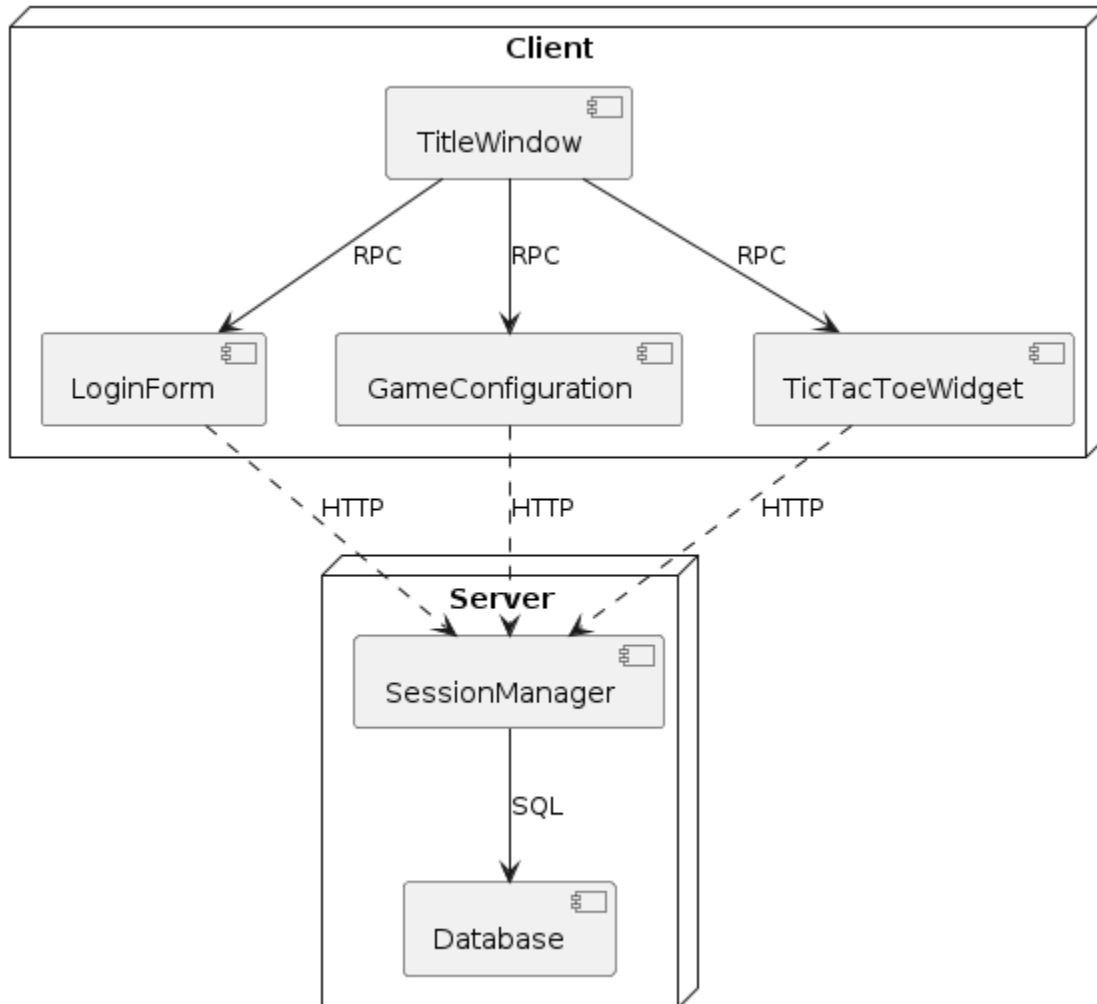


Figure 7: Deployment Diagram

Comment: This diagram depicts the physical deployment of the system, showing how the application components are distributed across client and server nodes.

9. Conclusion

The diagrams included in this SDS report provide a comprehensive view of the Tic Tac Toe application, covering functional requirements, dynamic behaviors, state transitions, component interactions, deployment topology, and class structure. These diagrams serve as a guide for developers and stakeholders to understand the design and architecture of the system.