

Chess Game System Design Document

Web-Based Chess Application

Document Version 3.0

Product Name: CheckMate

Last Updated: December 2024

Author: Jaimin, Werner, Flavio, Monica

Prepared By: Jaimin



Table of Contents

1. Introduction

- 1.1 Purpose
- 1.2 Scope
- 1.3 System Overview

2. Architectural Overview

- 2.1 Frontend Layer (View)
- 2.2 State Management (Model)
- 2.3 Backend Services (Controller)

3. Component Architecture

- 3.1 Core Components
- 3.2 Service Layer
- 3.3 State Management
- 3.4 Data Flow

4. Component Specifications

- 4.1 Frontend Components
- 4.2 Game Logic Components
- 4.3 Service Components

5. Class Responsibilities and Collaborations

- 5.1 Game Management Classes
- 5.2 Chess Piece Classes
- 5.3 Board Management Classes
- 5.4 Player Interface Classes

1. Introduction

1.1 Purpose

This document details the system design for a web-based chess application implemented using React and Java as backend. It serves as a comprehensive guide for development teams and collaborators.

1.2 Scope

The system consists a full-featured chess game with:

- Real-time multiplayer functionality
- Guest play mode
- Internationalization support
- Move validation
- Game state management

1.3 System Overview

The application follows a modified MVC pattern adapted for React, incorporating modern web development practices and microservices architecture.

2. Architectural Overview

2.1 Frontend Layer (View)

- User Interface Layer
 - Main container for React components
 - Modular component architecture
 - Responsive design implementation
- Core Components
 - Chess Component (Main controller)
 - Language Selector
 - Board Component
 - Piece Component

2.2 State Management (Model)

- Game State Management
 - React useState implementation
 - Immutable state updates
 - Centralized state control
- Language State
 - Translation key system
 - Dynamic text rendering
 - Language preference management

2.3 Backend Services (Controller)

- REST API Layer
- Game Logic Service
- Translation Service
- Database Integration

3. Component Architecture

3.1 Core Components

Detailed breakdown of main system components:

1. ChessGame Component
 - Game state management
 - Board display
 - Piece movement
 - Game initialization
 - Board data conversion
2. Chess Service
 - API communication
 - Session management
 - Move validation
 - State synchronization
3. Language Provider
 - Context management
 - Translation functions
 - Language switching

3.2 Service Layer

- RESTful API design
- Microservices architecture
- Scalable backend services

3.3 State Management:

- Centralized state management
- Immutable state updates
- Clear data flow patterns

3.4 Data Flow: •

Unidirectional data flow

- Event-driven architecture
- Clear service boundaries

4. Component Specifications

4.1 Frontend Layer (View):

- User Interface Layer: Main container for all React components • Components:
 - Chess Component: Main game controller
 - Language Selector: Handles language switching
 - Board Component: Manages chess board state □
 - Piece Component: Individual chess piece logic

4.2 State Management (Model):

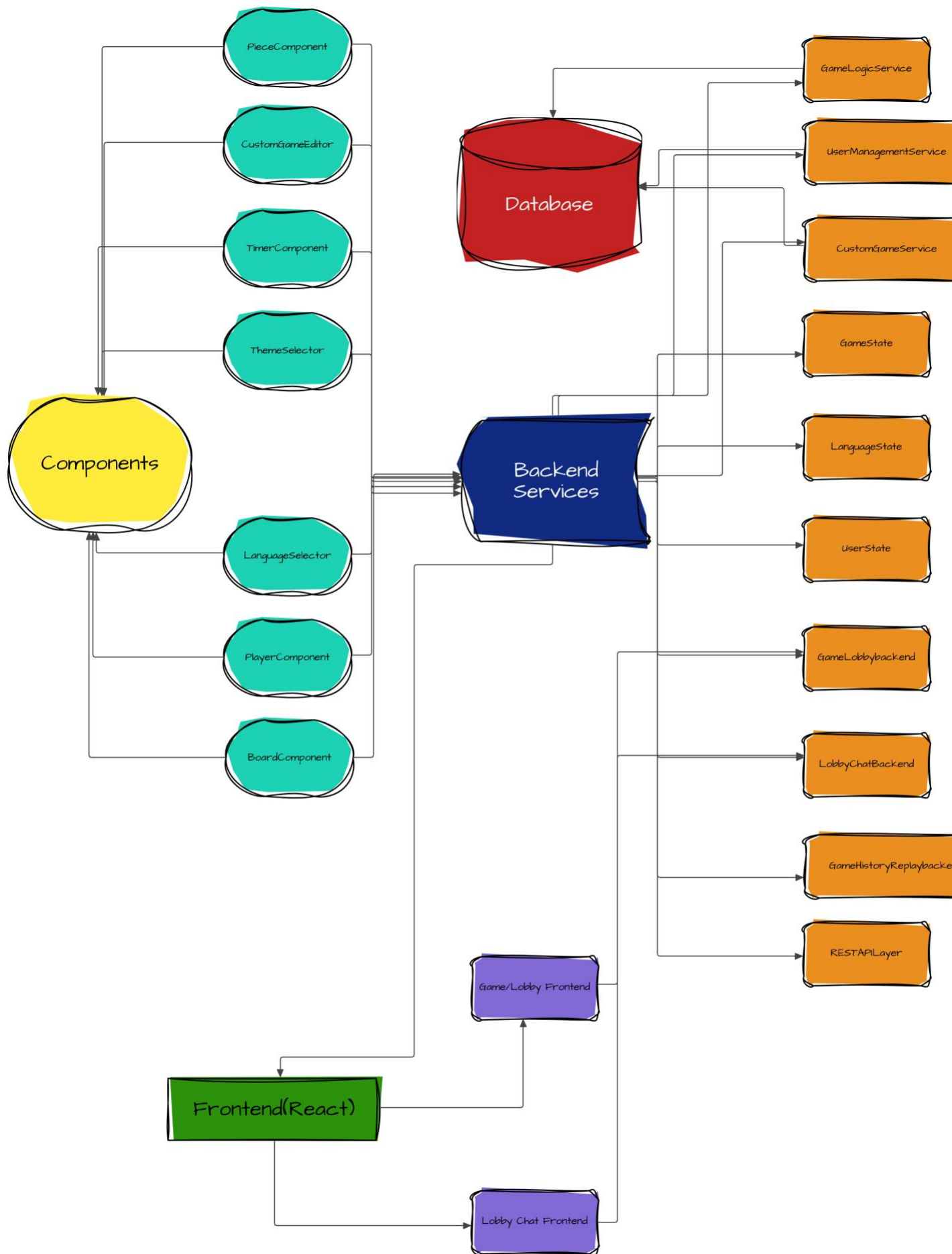
- Game State: Manages chess game state using React useState
- Language State: Handles language preferences and translations

4.3 Backend Services (Controller):

- REST API Layer: Handles communication between frontend and services
- Game Logic Service: Implements chess rules and validations
- Translation Service: Manages multilingual support
- Database: Stores game states and language data

5. CRC cards for Frontend and Backend Chess Logic

Software Architecture Design Diagram:



CRC Card Builder Page

CRC Card 1

ChessGame

Card Name	ChessGame		
Superclasses			
Subclasses	React.Component		
Responsibilities	Manages game state and board Handles piece selection and mov Updates board display Manages captured pieces Handles game initialization Updates board periodically Converts board data between for Renders chess board UI	Collaborators	ChessService LanguageProvider Board State Manager

CRC Card 2

ChessService

Card Name	ChessService
-----------	--------------

Card Name	BoardStateManager
Superclasses	

Superclasses			
Subclasses			
Responsibilities	Handles API communication Starts guest game session Sends move commands Retrieves board state Manages game IDs	Collaborators	ChessGame Backend API

CRC Card 3LanguageProvider

Card Name	LanguageProvider
-----------	------------------

Card Name	GameStateManager		
Superclasses			
Subclasses			
Responsibilities	Tracks game session state Manages player IDs Handles error states	Collaborators	ChessGame ChessService

Superclasses			
Subclasses			
Responsibilities	Provides language context Manages language state Provides translation function Switches between languages	Collaborators	ChessGame Translation Service

CRC Card 4 BoardStateManager CRC Card 5 MoveHandler

Card Name	MoveHandler		
Superclasses			
Subclasses			
Responsibilities	Validates piece selection Processes move attempts Updates piece positions Tracks captured pieces Manages player turns	Collaborators	ChessGame ChessService

CRC
Card 6

Subclasses			
Responsibilities	Initializes board array Converts piece representations Manages piece colors and types Handles captured pieces Converts API responses to UI for	Collaborators	ChessGame ChessService
	Updates game status Maintains current player turn		

UIRenderer

Card Name	UIRenderer		
Superclasses			
Subclasses			
Responsibilities	Renders chess board grid Displays piece images Shows captured pieces Handles square highlighting Displays game status Shows current player	Collaborators	ChessGame LanguageProvider

CRC Card 7

GameStateManager

CRC Card Builder Page

CRC Card 1

Pawn

Card Name	Pawn		
Superclasses	ChessPiece		
Subclasses			
Responsibilities	Represents each pawn piece Keeps track of moves made Defines its legal move Returns pathway of move	Collaborators	ChessBoard

CRC Card 2

Rook

Card Name	Rook		
Superclasses	ChessPiece		
Subclasses			
Responsibilities	Represents each rook piece Keeps track of moves made Defines its legal move Returns pathway of move	Collaborators	ChessBoard

CRC Card 3 Knight

Card Name	Bishop		
Superclasses	ChessPiece		
Subclasses			
Responsibilities	Represents each Bishop piece	Collaborators	ChessBoard

	Keeps track of moves made Defines its legal move Returns pathway of move		
Responsibilities	Represents each Knight piece Keeps track of moves made Defines its legal move Returns pathway of move	Collaborators	ChessBoard
Card Name	Knight		
Superclasses	ChessPiece		
Subclasses			

CRC Card 4 Bishop CRC Card 5 Queen

Card Name	Queen		
Superclasses	ChessPiece		
Subclasses			
Responsibilities	Represents each Queen piece Keeps track of moves made Defines its legal move Returns pathway of move	Collaborators	ChessBoard

CRC Card 6 King

Card Name	King		
Superclasses	ChessPiece		
Subclasses			

Card Name	ChessBoard

Superclasses			
Subclasses			
Responsibilities	Contains the board and pieces Makes move on board Checks if input move is valid Keeps track of Kings in Check Contains the logic of the board Prints board and state of board	Collaborators	Chess
Subclasses	Pawn Rook Knight Bishop Queen King		
Responsibilities	Contains info of color and type Contains its move history Gets pathway for move	Collaborators	ChessBoard Chess
Responsibilities	Represents each King piece Keeps track of moves made Defines its legal move Returns pathway of move	Collaborators	ChessBoard

CRC Card 7

ChessPiece (abstract)

Card Name	ChessPiece (abstract)
Superclasses	

CRC Card 8
CRC Card 9

ChessBoard
Chess

Card Name	Chess
Superclasses	
Subclasses	

CRC Card 10

Controller

Card Name	Controller		
Superclasses			
Subclasses			
Responsibilities	Creates the chess game Creates the player objects Interacts between player and gam	Collaborators	Player Chess

Card Name	Move		
Superclasses			
Subclasses			
Responsibilities	Represents coordinates on board Contains int row and column	Collaborators	ChessPiece Pawn Rook Knight

CRC Card 11 Move CRC Card 12				Bishop Queen King ChessBoard Chess Controller
	Responsibilities	Links ChessBoard and Controller Takes care of moves handling Keeps track of players turn Checks if game has ended Takes care of game state Gets winner of the game	Collaborators	ChessBoard Controller

Placeholder

Card Name	Placeholder			
Superclasses				
Subclasses				
Responsibilities	Represents tiles on the board Contains chess piece or an empt	Collaborators	ChessBoard ChessPiece	

CRC Card 13

Player

Card Name	Player			
Superclasses				

Subclasses			
Responsibilities	Reads inputs from players	Collaborators	Controller Chess

CRC Card 3

Theme Provider

Card Name			
Superclasses			
Subclasses			
Responsibilities	Provides theme context Manages theme state Provides coloring function Switches between themes	Collaborators	ChessGame Translation Service

CRC Card Builder Page

CRC Card 1

TimerMode

Card Name	TimerMode		
Superclasses			
Subclasses			
Responsibilities	Manage timer state Updates player times Handles time limit selection Manages game over by time	Collaborators	ChessGame GameState

CRC Cards Game Lobby and Lobby Chat

CRC Card 1

ChatBox

Card Name	ChatBox		
Superclasses			
Subclasses			
Responsibilities	To manage the user chat interfac	Collaborators	

CRC Card 2

ChatService

Card Name	ChatService		
Superclasses			
Subclasses			
Responsibilities	To provide a means to write and m	Collaborators	

ChessBoardComponent

CRC Card 1

ChessBoard

Card Name	ChessBoard		
-----------	------------	--	--

Superclasses	React		
Subclasses	None		
Responsibilities	Renders chess Board with current state Converts the board into 2d array Renders chess pieces on the board	Collaborators	Chess Pieces Interface Chess Board Square

CRC Card 2HistoryReplayComponent

Card Name	HistoryReplayComponent		
Superclasses	React, useState, useEffect		
Subclasses	None		

Responsibilities	Manages State of Game History Handles User Interactions Buttons for Play, Pause, Rewind, Fast Forward Renders chess board with current state Renders the list of Games for replay	Collaborators	ChessBoard Component Theme provider Game History Database Replay Controls
------------------	---	---------------	--

CRC Card Builder Page

CRC Card 1

Pawn

Card Name	Pawn		
Superclasses	ChessPiece		
Subclasses			
Responsibilities	Represents each pawn piece Keeps track of moves made Defines its legal move Returns pathway of move	Collaborators	ChessBoard

CRC Card 2

Rook

Card Name	Rook		
Superclasses	ChessPiece		
Subclasses			
Responsibilities	Represents each rook piece Keeps track of moves made Defines its legal move Returns pathway of move	Collaborators	ChessBoard

CRC Card 3

Knight

Card Name	Knight		
Superclasses	ChessPiece		
Subclasses			
Responsibilities	Represents each Knight piece Keeps track of moves made Defines its legal move Returns pathway of move	Collaborators	ChessBoard

CRC Card 4

Bishop

Card Name	Bishop		
Superclasses	ChessPiece		
Subclasses			
Responsibilities	Represents each Bishop piece	Collaborators	ChessBoard

