Implementation Notes:

- 1. Component Architecture:
 - Main ChessGame component acts as container
 - Modular services for API and language
 - Separate concerns for state management
- 2. State Management:
 - Uses React useState for game state
 - Periodic board updates with useEffect
 - Centralized state in ChessGame
- 3. API Integration:
 - RESTful service calls
 - Guest game functionality
 - Move validation on server
 - Board state synchronization
- 4. Internationalization:
 - Language provider integration
 - Translation key system
 - Dynamic text rendering
- 5. Key Features:
 - Guest play mode
 - Real-time board updates
 - Piece capture tracking
 - Move validation
 - Multi-language support

The architecture follows a modified MVC (Model-View-Controller) pattern adapted for a React-based web application.

- 1. Frontend Layer (View):
 - o User Interface Layer: Main container for all React components
 - o Components:
 - Chess Component: Main game controller
 - Language Selector: Handles language switching
 - Board Component: Manages chess board state
 - Piece Component: Individual chess piece logic

- 2. State Management (Model):
 - o Game State: Manages chess game state using React useState
 - o Language State: Handles language preferences and translations
- 3. Backend Services (Controller):
 - o REST API Layer: Handles communication between frontend and services
 - o Game Logic Service: Implements chess rules and validations
 - o Translation Service: Manages multilingual support
 - o Database: Stores game states and language data

Key Architectural Features Covered:

- 1. Component-Based Architecture:
 - Modular components for reusability
 - Clear separation of concerns
 - Hierarchical component structure
- 2. State Management:
 - o Centralized state management
 - o Immutable state updates
 - Clear data flow patterns
- 3. Service Layer:
 - o RESTful API design
 - Microservices architecture
 - Scalable backend services
- 4. Data Flow:
 - Unidirectional data flow
 - o Event-driven architecture
 - Clear service boundaries

