OnlineChess

Software Requirements Specification (SRS)

Vansh Gupta
Jay Patel

Date: 11/28/23

Table Of Contents

1. Introduction

1.1 Purpose

The purpose of this SRS document is to go through the description, features, and user classes of the OnlineChess application as well as specify and provide a proper understanding of the function, non-functional, and user interface requirements of the online chess application for stakeholders involved in the development as well as the use of the application.

The OnlineChess application consists of a **ChessServer** and **ChessClient**

ChessServer - Manages the chess game database and controls any and all aspects of the client-side interactions between users. It allows for game creation, real-time gameplay, messaging between users, connecting to the internet, account creation, as well as pausing and resuming games and being able to save and store the game data for later retrieval.

ChessClient: Installed on users' devices and connects to the ChessServer. It can load game data from the ChessServer and allow users to play with one another or be able to start new games. The ChessClient also allows users to create and log into their accounts and also be able to fully customize their profile and their chess game looks.

1.2 Scope

The OnlineChess application is a digital platform designed to provide an interactive chess experience for users globally. The scope of the application involves providing a virtual environment that copies the real-world experience of a chess game, where two players are able to play against each other from anywhere in the world.

OnlineChess is designed for three main user groups: The casual players who play friendly matches, the competitive players who play to advance their chess skills, and the normal players who play to have fun and enjoy spending time doing something new. Users will have the ability to personalize their gaming experience, handle their profiles, and be able to message other users within the game. The functionalities of OnlineChess will be made very flexible to accommodate players of all different skill levels and preferences so it gives a good overall experience.

1.3 Audience

The primary audience includes the development team, project managers, stakeholders, and quality assurance testers.

Development Team: For developers, the document provides detailed descriptions of system functionality and architecture requirements that will guide coding and implementation. It includes a product description to give them background on the product, necessary features that will be included in the product, as well as specific requirements which include the function requirements, non-functional requirements, and the user interface requirements.

Project Managers: For project managers, the document gives them a good look at the resource allocation of the application as well as gives them the opportunity to make a plan in order to maximize efficiency and be able to have proper scope management

Stakeholders/Investors: For stakeholders, which includes investors and future product owners, the document outlines the vision of the product, as well as the features they can see in the product and samples of a potential user interface.

Quality Assurance Testers: The documents can be used by Quality Assurance testers as a benchmark for making and testing out their test cases, ensuring that all functionalities that are written in the document are validated and whatnot

1.4 Overview

The Software Requirements Specification document shows the development process of the OnlineChess application. It is organized into **three sections**, the introduction, overall description, and specific requirements.

The **Introduction** section sets the stage for the OnlineChess application, establishing a purpose, scope, as well as intended audience of the SRS document.

Following the introduction, the **Overall Description** section shows a top-down view of the product, including a thorough product description that sums up the OnlineChess experience. This section describes the product's features, potential user base, operating environments, and any assumptions or constraints that may impact design and functionality.
The final section on **Specific Requirements** is the core of the document. It is a collection of all the system's requirements, categorized into functional, non-functional, and user interface specifications. These include detailed descriptions of system

operations, benchmarks of performance, security protocols, and the user interaction model. This section is intended to provide a comprehensive and insightful checklist of requirements that will direct the development team in creating a product that aligns with the needs of the user and the expectations of the stakeholder.

1.5 Reference

IEEE Std 830-1998 Recommended Practice for Software Requirements Specifications

2. Overall Description

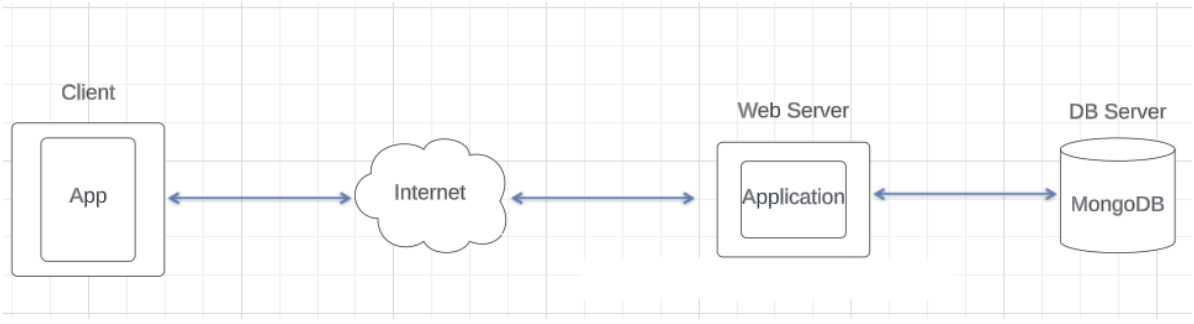
2.1 Product Description

OnlineChess is an application designed to recreate the classic chess experience for two players in a digital environment. It is not software for playing chess but rather it is an application that creates a virtual setting where users are able to play a game of chess with one another without being physically with each other and without the requirement of having physical chess pieces. The OnlineChess application includes a graphical user interface that represents a chessboard, where users are able to move pieces and only make valid moves just like how it is in any regular chess game. This virtual chessboard updates player moves in real-time, giving the user an even more realistic feeling environment. In OnlineChess, it follows all the same rules and guidelines as any normal game of chess, with players winning either through resignation by the other player, through checkmate, or through the other user running out of time during timed chess games

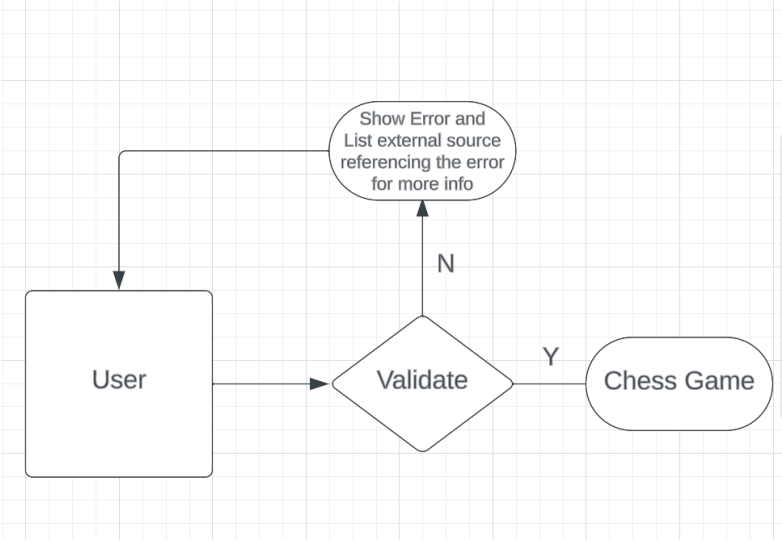
2.2 Product Perspective

OnlineChess is controlled by two major components, the ChessClient and ChessServer. The ChessClient is the application itself which the user interacts with directly and the ChessServer is the connection in the background that performs all of the actions on the chess client.

Below is a simple look at how this relationship will take place in the OnlineChess application for when a user makes a move and for the move to get validated.



This system above is at the client level



This is how the system would be at the server level which pushes the results back to the chess game client once it is processed

The OnlineChess application architecture takes on a client-server model, with a backend infrastructure that will be in charge of handling data transmission, user management, and gameplay logic. Data persistence is managed through a MongoDB database server, ensuring the security and integrity of data from users.

2.3 Product Features

There are several major features of OnlineChess, which all have a key role in the requirements of the application.

2.3.1 Easy-to-Use Interface: Within the chess application, it will provide a proper and clean-looking user interface that will be easy to use by users of all ages, levels, and knowledge of chess. It will be very similar to a physical chess game, and allow users to be able to drag their pieces and drop them where they want to go.

2.3.2 Customization: With customization, it allows the ability to not only customize the look and layout of their chess board and pieces but also customize user profiles. There are several different board and piece options, with different patterns and colors and much more.

2.3.3 Account Login/Registration: Through using the application, it allows for the option to register for an account and then down the road be able to login to the account to view past games and game history as well as be able to add people to play with.

2.3.4 Multiple Game Modes: OnlineChess will allow for the ability to play several game modes, including timed chess games where you are able to set a timer for however long you want the game to be or untimed chess games where you are allowed to take as long as you want to play or even custom games where you can customize pieces and customize how the board setup is in case you wanted to practice.

2.3.5 Real-Time Moves: In OnlineChess, it allows for real-time updates in the playing field with the combined addition of move validation so the experience can be as similar as it can

2.3.6 In-Game Hints: During a game, if a user is stuck for a long time on a move, they will have the ability to request a hint, which the application will be able to analyze the state of the game and use strategies to give a potential next good move

2.3.7 In-Game Messaging: An in-game chat feature will allow players to be able to communicate with one another while playing the game, which brings a social aspect to the game of chess.

2.3.8 Pause Game: There will be flexibility to be able to pause the game which can be very useful with the way chess is often played

2.3.9 Save Game for Later: Coupled with being able to pause the game, OnlineChess gives the option to be able to save a game. This can allow being able to open the game back up at a later time which can be very good because it makes it so the game can be played at a later time or multiple games can be played at the same time without the data from one game being lost.

2.3.10 Only perform Legal Moves: The system enforces only legal FIDE rules to be performed, ensuring that games are fair and stick to the traditions of chess, unless games are customized with custom chess pieces and layouts

2.3.11 Review Statistics: Upon compilation of a game, players have the ability to keep track of their win/loss count as well as review their game and see where they could have potentially gone wrong or right in order to improve on their chess skills

2.3.12 View Tutorials: The OnlineChess application includes several built in tutorials on chess that vary in experience levels. They can be useful, similar as reviewing games, to improve chess skills

2.3.13 Add Friends: Users can add their friends to be able to play games with them or request games from them whenever they want to.

2.4 User Classes and Characteristics

OnlineChess consists of **Physical** actors which are the users who use the application and **System** actors which are the **Chess Client**, **Chess Server**as well as the **Chess Database**

Physical Actors:
Chess Player: This is the primary user of the OnlineChess application who engages in chess matches using the client interface. Each chess player represents an individual user who connects to the platform to play games, interact with other players, and use the features offered by the application which were mentioned above.

Chess Enthusiast: The chess enthusiast is a user who might engage with the platform more deeply, through reviewing games and studying game data. They represent users who explore the capabilities of the OnlineChess application, beyond just being able to play chess.

System Actors:
Client: The client is the front-end system where users interact with the platform. It handles gameplay, user interactions, and other client-side features, ensuring a smooth and responsive experience for the players.

Server: The server is the back-end system that manages game logic, user connections, and stores game data.

Database: The database handles the storage and retrieval of game records, user profiles, and analytics. It allows players to review past games, track their progress, and access several other features which can be very useful.

Goals of the Actors:
The primary goal of the Chess Player, the core physical actor, is to enjoy playing chess, improve their skills, and engage with the chess community through the OnlineChess platform. The Chess Enthusiast aims to utilize the advanced features of the platform for deeper engagement with the game, such as analyzing matches, setting up educational content, or organizing competitive events. The system actors (Client, Server, and Game Database) work in unison to facilitate a seamless, engaging, and educational experience for all users of the OnlineChess application.

2.5 Operating Environment

The application will be supported by mobile phones. Since IOS and Android are the leading operating systems in mobile technologies, they will be used as the main operating environments.
IOS 10 and 10+ would be supported by the Online Chess application and Android 4.4+(API level 19) with a Rokt framework which would support transaction systems.

2.6 Assumptions and Constraints

A**ssumptions** with the OnlineChess application are that the users have a basic understanding and basic knowledge of chess. Along with a basic understanding of chess, they need to have access and proficiency to a client device that will be able to run the application as well as an internet connection to be able to use the application. Without an internet connection, the user will not be able to connect to the ChessServer through their client.

Constraints behind the OnlineChess application are mainly technical and have to do with network connectivity. Without a properly established and working network connection, this will lead to slow loading times for several aspects of the application which can cause performance issues. There are also constraints with data management and security because having proper security features for users and their data, can have an impact on the system.

3. Specific Requirements

3.1 Functional Requirements:

3.1.1 User Registration & Login

FR-1. User can register a new account
FR-2. User can log-in with a guest profile
FR-3. User can log-in with an account

3.1.2 Tutorials & Home Screen Features

FR-4. User can view tutorials
FR-5. User can start new games
FR-6. User can resume old games

3.1.3 Player Profiles & Adding Friends

FR-7. User can find players using their profile ID and send a friend request
FR-8. User can accept or reject friend requests
FR-9. User can view their stats, ranking, and other info about themselves
FR-10. User can search for players using their profile ID and view their game stats

3.1.4 In-Game Features

FR-11. User can add a player to play against
FR-12. User can send a message
FR-13. User can read the received message
FR-14. User can make a legal move during their turn
FR-15. The system enforces the rules of chess set by FIDE(Fédération Internationale des Échecs)
FR-16. If the user quits the game is saved automatically
FR-17. If user is online and the timer runs out the other player wins the game

3.1.5 Security

FR-18. The system has security measures to prevent breaches of data and user privacy
FR-19. The system can securely process a user's interaction with the system.

3.2 Non-Functional Requirements

3.2.1 Reliability

NFR-1. The system is reliable and doesn't crash.
NFR-2. The system doesn't have system failures when a player is trying to make legal moves or access features of the app
NFR-3. The system has very low downtime when new updates are added or the

system is repaired

3.2.2 Availability

NFR-4. The system is available to use at all times except for expected downtime .

3.2.3 Scalability

NFR-5. The system can be upgraded to accommodate more users and/or make the experience seamless

3.2.4 Usability

NFR-6. The system is accessible to users with high eye power and sensitive eyes

NFR-7. The system is usable by users not very proficient in computers and has prompts to help them access features

3.2.5 Performance

NFR-8. The number of users can be a lot and the system should still work seamlessly

NFR-9. The moves and screen changes should be smooth and fast

3.2.6 Portability

NFR-10. The system can be accessed from multiple devices and multiple locations

3.2.7 Modifiability

NFR-11. The system can be easily modified and new features can be added

3.2.8 Supportability

NFR-10. The system can be accessed using a device that can access the internet and has a UI like Mac, Windows, iOS, and Android.

3.3 User Interface

1. Start Game

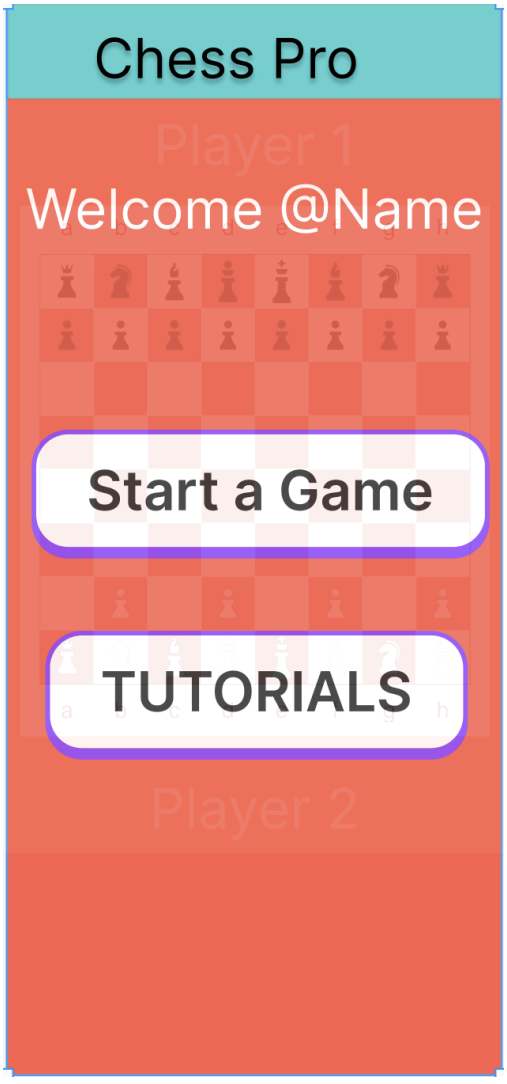
Usage Scenario: The users can start a new game

Pre-condition: Both Users are connected and ready to play, Users have a stable internet connection

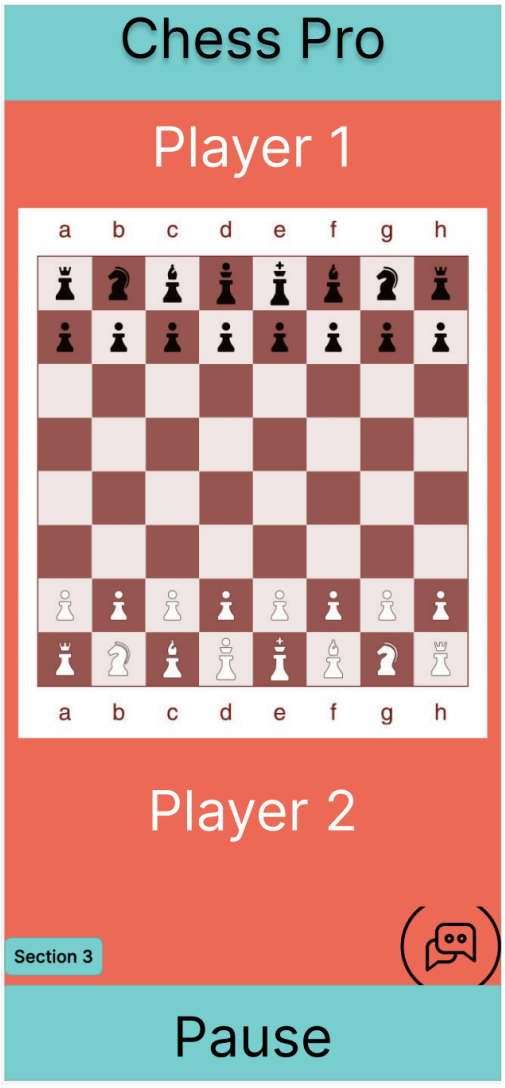
Post Conditions: A game session is created

Base Case of Actions: The user chooses a game mode to play, The user invites the second player, and the second player successfully joins the game, When a user clicks start to game the users are guided to the game board

Screen 1:



Screen 2:



2. Save Game

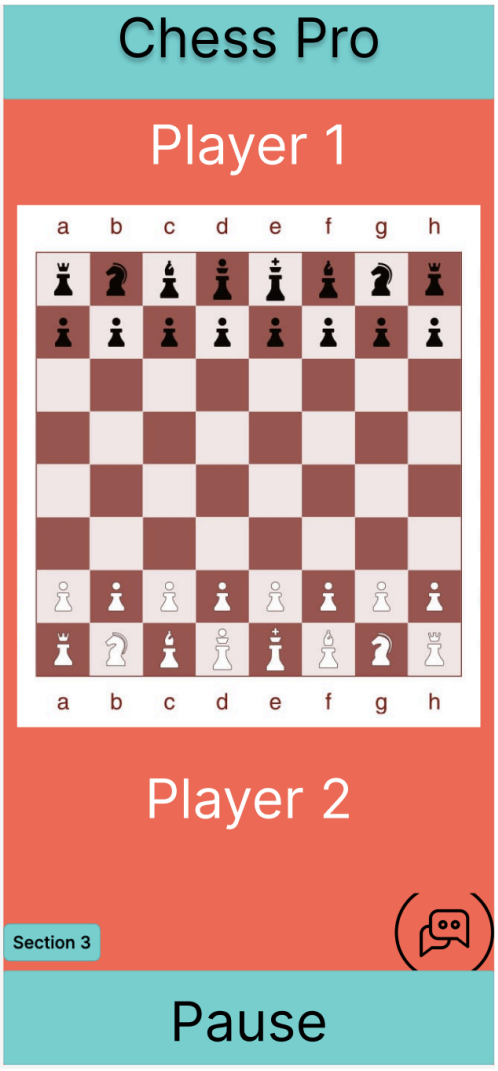
Usage Scenario: The users can pause/save a game while in the middle of it

Pre-condition: Both users are in an active game session, Users have stable internet, One player attempts to pause the game

Post-condition: The game is paused and saved for both players and moves cannot be made

Base Case of Actions: The user clicks on a pause button, The status of the game is saved and the timer (if applicable) stops, The game can start back up whenever both users are ready again.

Screen 1:



Screen 2:

3. Chat During Game

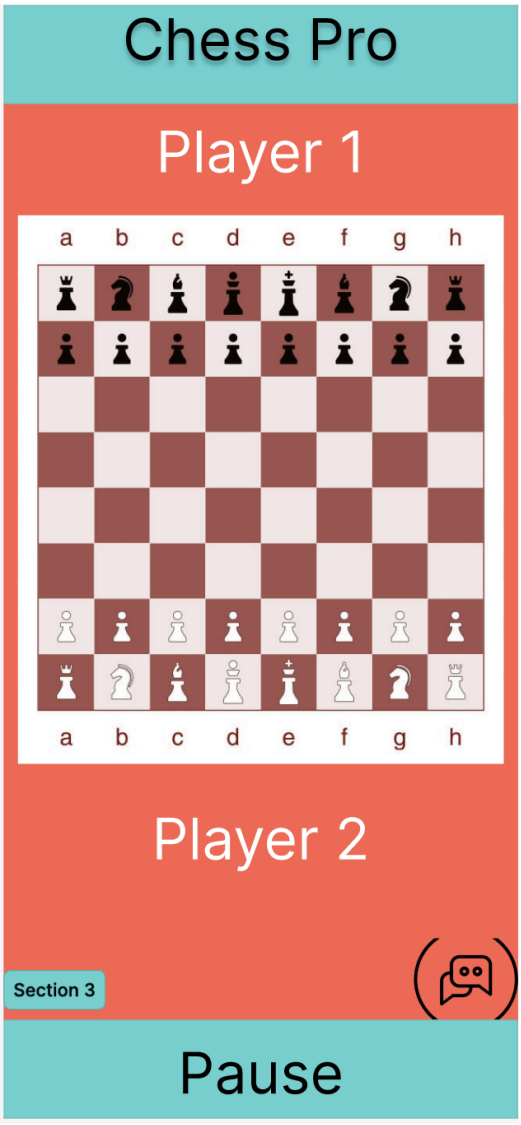
Usage Scenario: The players can chat during the game

Pre-condition: Both Users are playing the game, and the Users have a stable internet connection

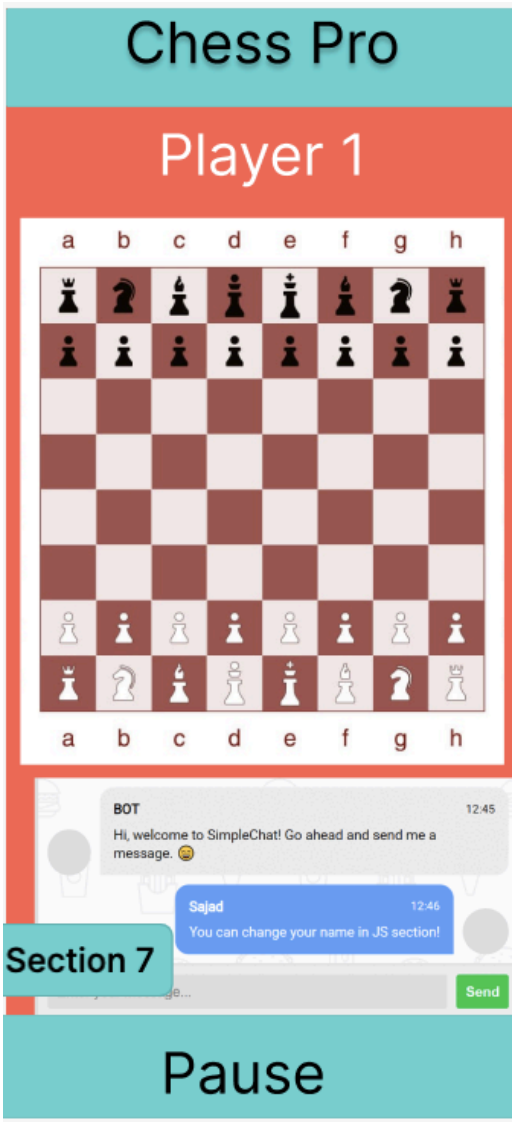
Post Conditions: The other user receives the message in the same language as when it was sent

Base Case of Actions: The user opens the chat bar, types a message and presses send, The second user has a notification and the chat bar has a number appear for unread messages

Screen 1:



Screen2:



4. ViewTutorials

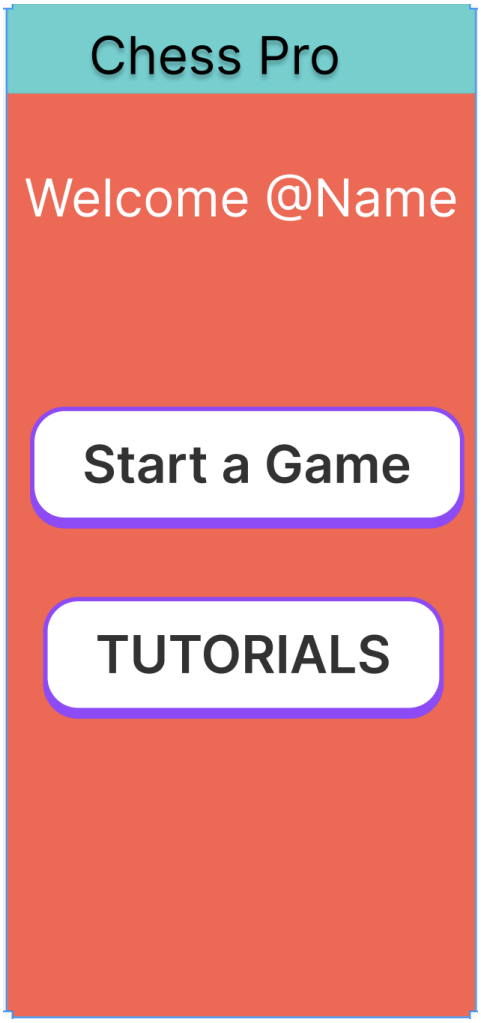
Usage Scenario: The user can look at tutorials

Pre-condition: The user is on the OnlineChess application, User have a stable internet, User is ready to play

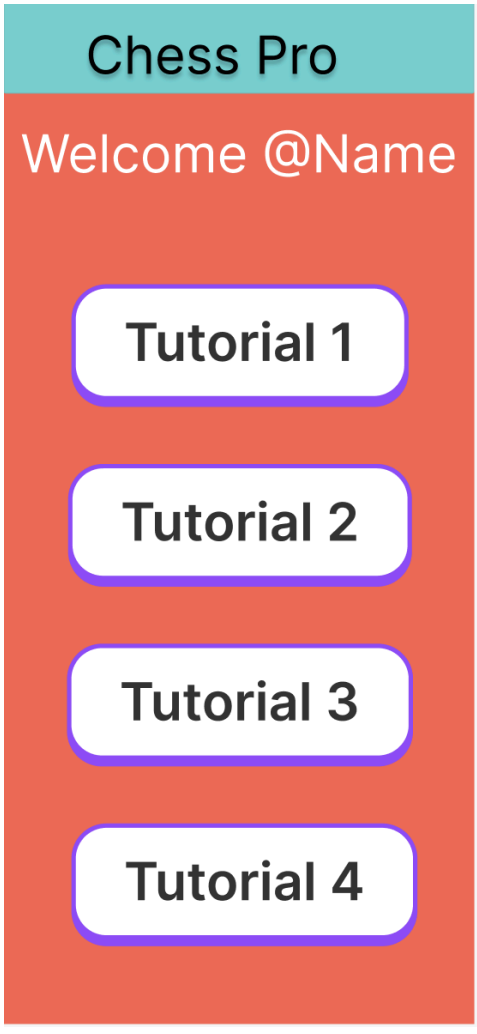
Post-condition: User knows about set-piece rules and how the game works

Base Case of Actions: The user selects view tutorial, Applications shows multiple tutorials, User plays tutorials to learn rules, set piece movements and general working of applications and how to access it

Screen 1:



Screen 2:



Requirements:
Pagination: For tutorials, friends lists, and moves, pagination shall be implemented.

Distribution Map: The distribution map should enable users to pan, zoom in, and zoom out on the screen to help solve Usability NFR.

Filter Building Interface: During the creation of a filter to search for users, users should be provided with real-time counts of matching occurrences before initiating the search.