

# ABSTRACT

Chess, a game originated in India and active since 1475 can be argued to be the most popular and interesting board game in the world. The game, which currently has over 700 million players, has seen both- the game ending in two moves and a match that lasted over 20 hours. This already creative game with its infinite number of moves has been further innovated with aspects like blindfold gameplay. Hence, it is only fair that in the digital world, the board game has its digital place.

This project, Chessverse, intends to create a platform where chess enthusiasts can play, compete and enhance their skills. Chessverse would be an online platform that provides tutorials and gaming interface. We use HTML and CSS for the development of web pages.

In this project, we have developed a safe and fluid experience for players to compete both through solo mode and matches against friends. For solo gameplay, a player can compete against a bot or an AI(Stockfish). We plan to include an effective login system to prevent unauthorized access to the game and monitor the progress of all the users on the platform and a post-game analysis that educates the players on their mistakes and how to avoid them in the future. The end vision is to provide the user with an all-engaging chess environment where one can play, learn and socialize at the same time.

# TABLE OF CONTENT

S.NO.	TITLE
1	Introduction
2	Software Requirements and Specification
3	Project Plan
4	Designing Strategy
5	Simulation and Result Analysis
6	Conclusion
7	Future Scope
8	References

# INTRODUCTION

The chessverse is a software that provides a comprehensive platform for chess enthusiasts to play chess and enhance their skills through the various game modes provided.

It allows the user to login/signup- these pages are interconnected and also have a 'forgot password' option for the user's convenience. The login page leads to an index page that has several gameplay options that include local multiplayer, play against bot(for beginners) and play against AI(named Stockfish).

The primary goals of the software are:

- Comprehensive platform for chess enthusiasts to manage board games.
- Facilitating communication among members..
- Replaying moves to understand the flow of the game. This helps the user to understand all moves and develop skill.
- Premove option that allows the user to save their next move while the opponent takes their time. This can be beneficial to the player when they are short on time.
- Ensure a user-friendly and efficient experience for users.

The target audience for the software includes:

- People who are interested in chess and want to enhance their skills.
- Individuals who manage the software and want to oversee the software and its users.

The scope of the software includes:

- It manages the user registration and login of the user.
- It provides an insight of the performance of players and their strategies.

# SOFTWARE REQUIREMENT

## FUNCTIONAL REQUIREMENT

### 1. User Management

- User Registration and Login System
- Types of users: admin, member/guest

### 2. Payment gateway

- Subscription for premium features

### 2. Game Management

- Start and end chess games (online)
- Players can play with friends and accordingly choose the color of the chess piece .
- Choose gamemode- local multiplayer, against bot, loading games from PGN and against AI.
- Recorded moves for game preview.

### 3.Player Management

- The players can choose with whom they want to play.
- The individual who wants to invite a friend can send an invite link.

## INTERFACE REQUIREMENTS

The software will have a simple to use user interface Different level users will be able to access the variable interface.

- It will be available on the desktop .
- It will have responsive design based on the user's screen dimensions.

- The software would provide a consistent design and layout everywhere.

## PERFORMANCE REQUIREMENTS

- It will be made in a way to work fast and inputs respond quickly.
- This ensures that the software will be capable of managing the concurrent users.
- This means that the software will take very less time before it returns a response back to the user input.
- The software would have the ability to analyze and save data.

## DESIGN CONSTRAINTS

The software will be designed to meet the following constraints:

- The software will be built using a web-based framework .
- The software will use a relational database management system .
- The software will be deployed on a cloud-based platform.

## Non-Functional Attributes

The software will have the following non-functional attributes:

- The software will use encryption to protect user data and passwords.
- The software will be designed to be user-friendly and accessible.
- The payment gateway would be secure and user-friendly.
- The software will be able to handle it in case the number of users increases.

# PROJECT PLAN

## Project Overview

The software is a web-based application designed to provide a comprehensive platform for chess enthusiasts to play chess games as an individual or against a friend. If a player intends to play against a friend they can use the local multiplayer option and in case of solo gameplay, they can play against a bot or an AI. This project plan includes the scope, timeline, budget, and resources required to develop the software.

## Project Scope

The scope of the project includes:

- It manages the user registration and login of the user.
- It provides an insight of the performance of players and their strategies.
- It offers subscriptions for additional facilities.

The project timeline is divided into six phases:

### **Phase 1: Requirements Gathering**

- Analyze the attribute of the software and its functionality.
- Prepare a Software Requirements Specification (SRS) document.
- Fix the project scope, timeline, and budget.

### **Phase 2: Design**

- Create a design of the document outlining the architecture of the software, user interface, and the system architecture.
- Develop a prototype of the interface.
- Define the technical requirements needed for the software.

### **Phase 3: Development**

- Develop the software using a web-based framework .

- Implement the user management, game management and player management.

#### **Phase 4: Testing**

- Conduct various testings to ensure the software meets the requirements.
- Perform usability testing and take feedback.

#### **Phase 5: Deployment**

- Deploy the software on a cloud-based platform
- Set up the database.
- Perform final testing and quality check.

#### **Phase 6: Maintenance and Support (Ongoing)**

- Provide ongoing maintenance and support to ensure the software remains stable and secure.
- Fix bugs and issues reported by users.
- Implement new features and updates as required.

### **Project Resources**

The project requires the following resources:

- Project manager: 1 person
- Software developers: 1 people
- Technical writer: 1 person
- Front-end designer: 2 person

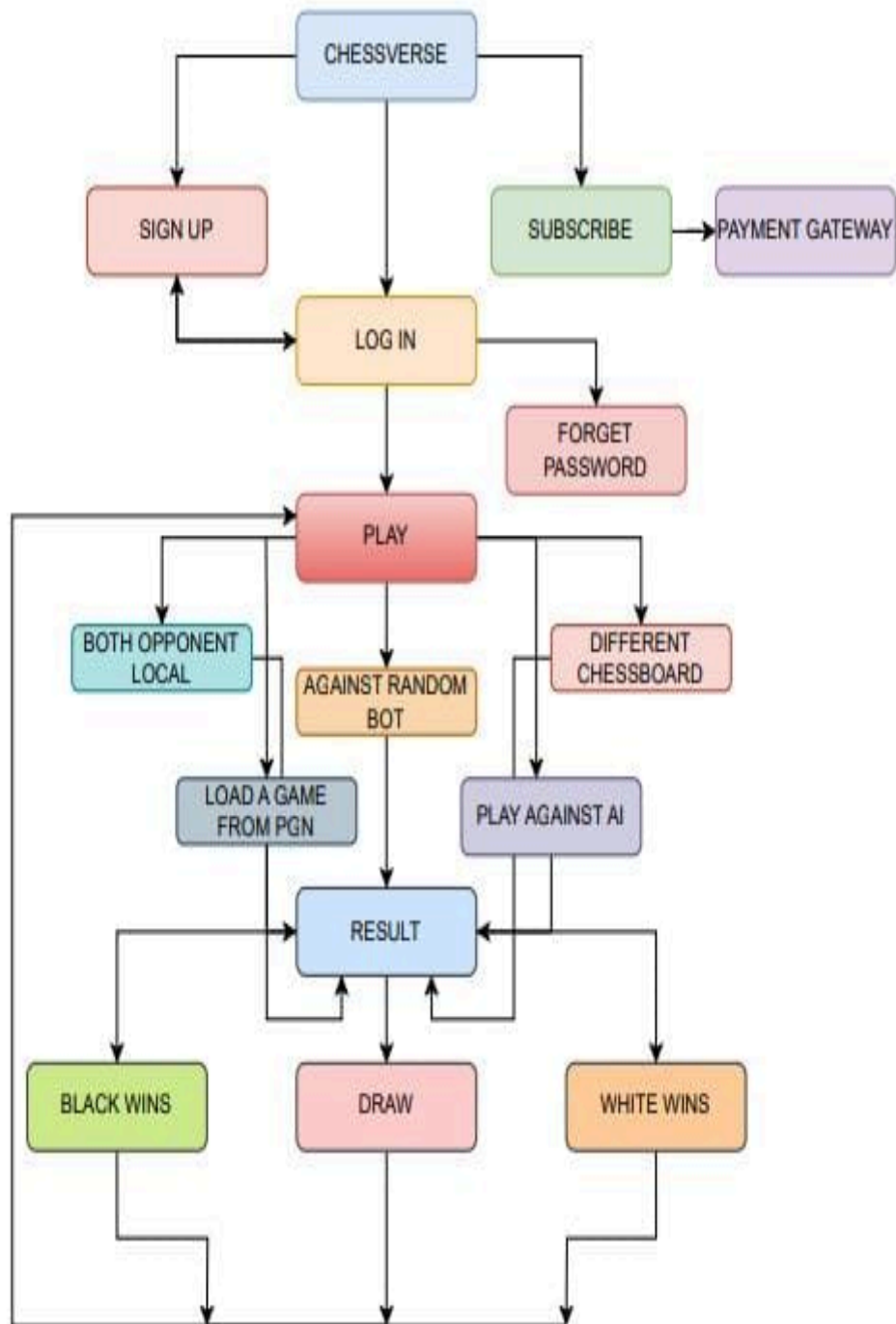
### **Project Risks and Assumptions**

The following risks and assumptions are associated with the project:

- Risk: Delays in development may impact the project timeline.
- Assumption: The development team has experience with web-based frameworks.
- Risk: Changes in requirements may impact the project scope and plan.

By following this project plan, we can ensure that the software is developed and deployed on time, and meets the requirements.

# PROCESS FLOW DIAGRAM





# DESIGNING STRATEGY

## Objectives and Goals:

We aimed to create a platform where chess enthusiasts can interact, play and improve their skills be it a beginner or an expert.

We aim for smooth gameplay on the several modes provided. The player can choose from these modes based on their skill level and will.

We plan on providing video or picture-based tutorials that can help a novice familiarize himself with the game. Furthermore, we are keen on building a user-friendly interface which provides player rankings.

## System Architecture:

First and foremost, the user has the following details or options in the homepage:

- Signup and Login: The user adds their data that leads them to the chess board, that is, the gaming interface.
- Subscribe: This option allows the user to become a member by paying a certain amount. The members then enjoy additional benefits.
- About: This option will tell the user about the site and its functionalities. It is yet to be fully developed.
- Contact us: Basic contact information that allows the user to communicate with the admin in case of any difficulty.
- Upcoming events

A successful login leads to the index page. Here, we see multiple options from where the user can choose according to their needs. These options include playing local multiplayer, play against bot(for beginners), play against AI(named Stockfish) and loading a game from PGN.

In case a user faces any trouble while logging in due to username and password issues, they can choose the option 'Forgot Password' provided in the login portal itself.

## Technology Stack:

We have used HTML for the development of web pages. This is because it provides enhanced multimedia support, offline capabilities and responsive web designs.

We also use CSS, which is a style sheet language used for specifying the presentation and styling of a document written in a markup language such as HTML. CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript.

Furthermore, Node.js which is a cross-platform, open source JavaScript runtime environment was used to write command line tools and server-side scripting by using JavaScript and SCSS, which is basically a more advanced and evolved variant of the CSS language was also used.

Lastly, the Stockfish github repository was used to export the chess AI to our software.

## Collaboration and Integration

Virtual Studio Code was used as the source code editor.

We intend to upload our work on GitHub that will provide us with a smooth workflow and collaboration as the complexity of this project increases.

# SIMULATION AND RESULT ANALYSIS

We have created a prototype that shows how a player can login and then choose among the different options to enter the game interface. We also have the Subscribe option that leads to a payment gateway. The purpose of simulating our results until now is to ensure that our system is performing as expected.

The different components of the system are:

- Home page of the Chessverse website: It provides some functional and some non-functional options.

The Sign Up option allows the user to create an account that will keep a track of their progress. The Sign up leads to the login portal.

If the user already has an account, they can directly login through the Log in option in the homepage.

Lastly, the Subscribe options directs the user to a payment gateway.

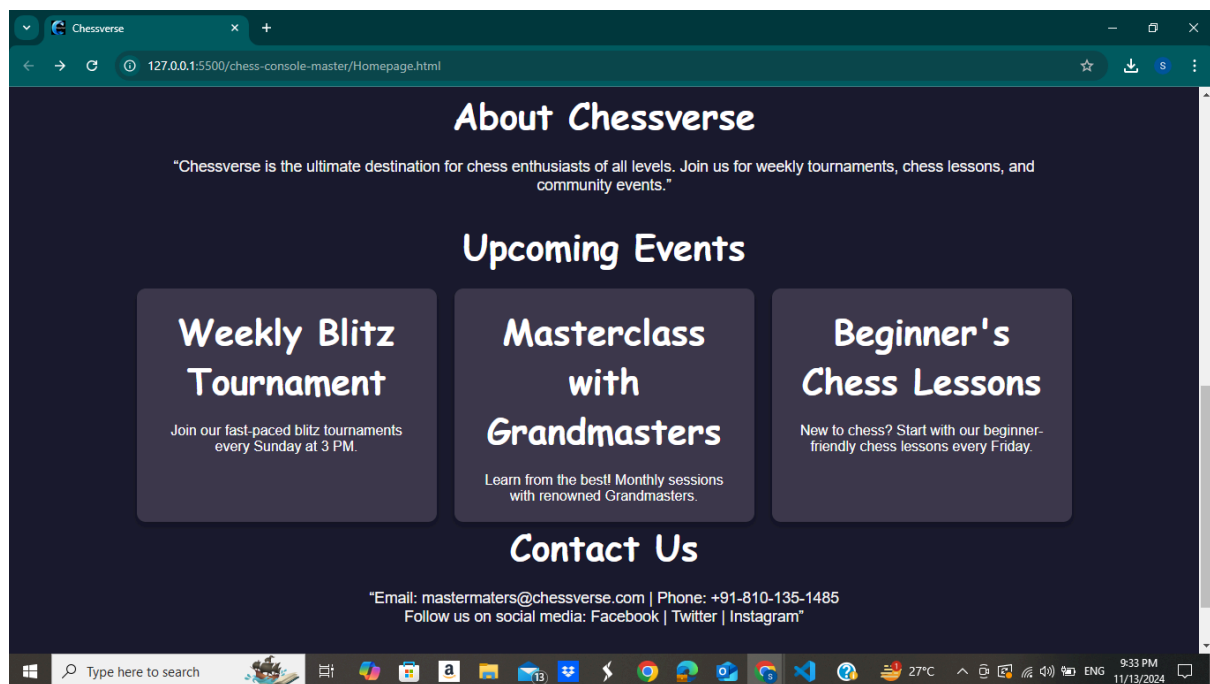


The homepage also has other options and designs as shown below:

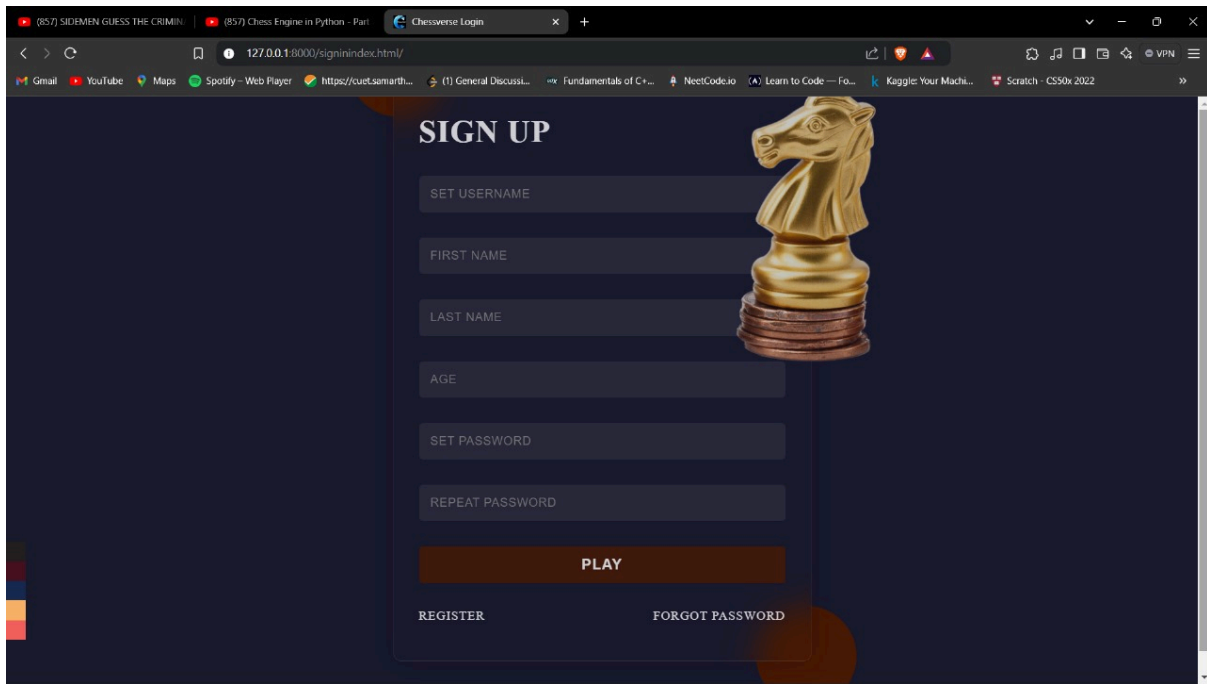
Website details are given in the About Us section.

Upcoming events are mentioned in a section where the user can get different chess tutorials.

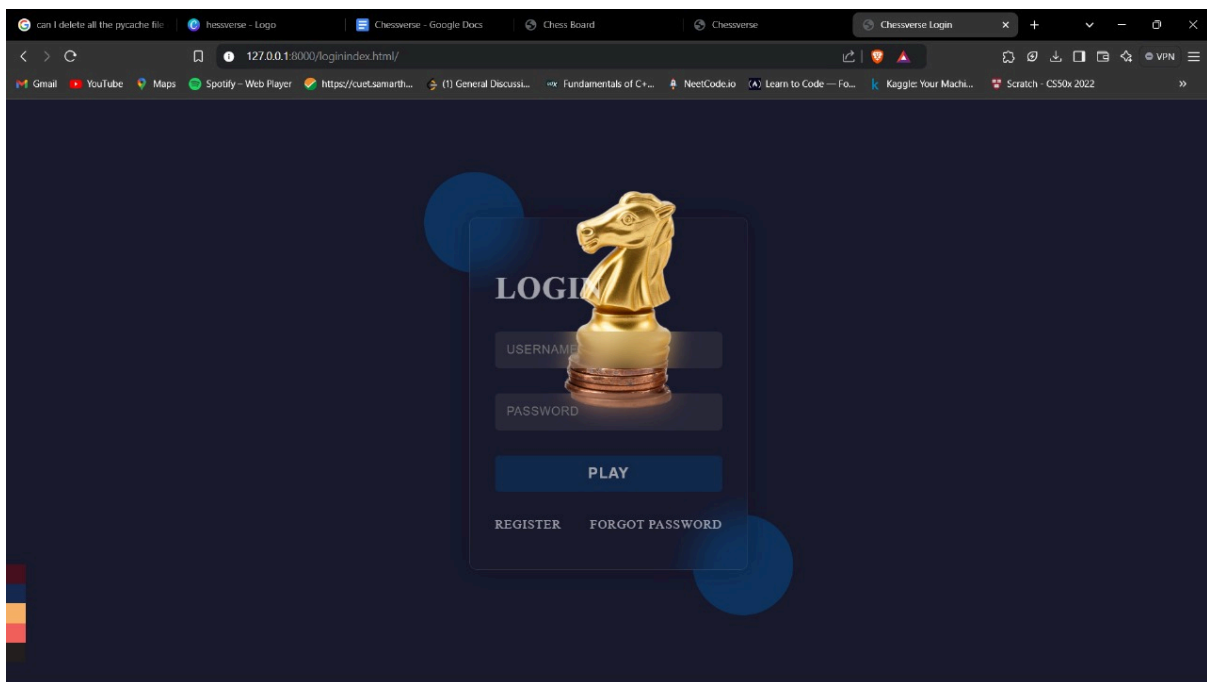
Lastly, the homepage has a Contact Us section where users can contact the admins in case of any difficulty.



- Sign up page: Takes detailed information like username, name, age and password from the user when initially creating an account. This page leads to the login portal.



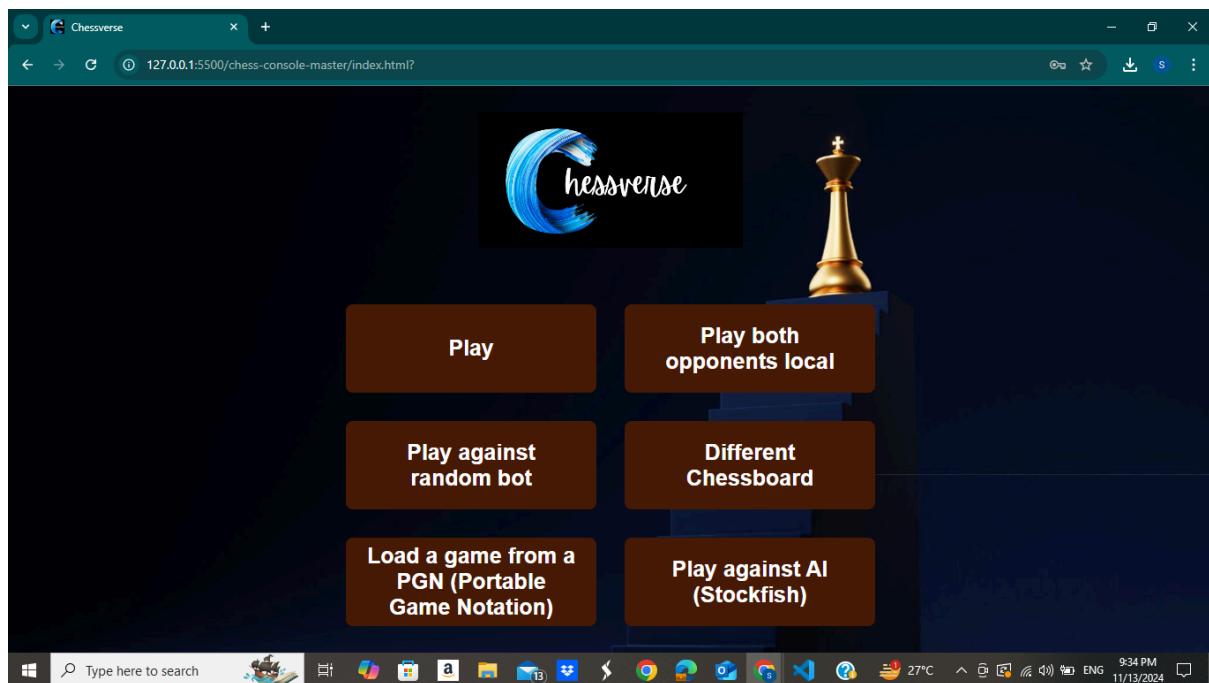
- Login portal: This takes the user input and directs them to the gaming interface. A user can directly go to the login page or follow it up from the sign up portal (for new users).



- Index page: This page shows six options from where the user can choose according to their needs.

The options are:

- A simple chess game
- Local multiplayer: Here, a player can rewatch the moves throughout the game which helps in self reflection and understanding blunders.
- Playing against a random bot: This is beneficial for new players as they can play at their own pace and learn about the movement of chess pieces.
- Loading the game through PGN: This mode can move the game to a certain point according to the pre mentioned notations.
- Play against AI: An AI named Stockfish is used so that players can practice at a higher level than a random bot.





Chessverse

127.0.0.1:5500/chess-console-master/examples/load-pgn.html

22. ♖f5+ ♜e8  
23. ♜d7+

♜♜♜♜♜ 7  
♜♜♜♜♜♜ 22

Player ■

The loaded PGN:

```
1.e4 e5 2.Nf3 Nc6 3.Bc4 Bc5 4.b4 Bxb4 5.c3 Ba5 6.d4 exd4 7.O-O  
d3 8.Qb3 Qf6 9.e5 Qg6 10.Re1 Nge7 11.Ba3 b5 12.Qxb5 Rb8 13.Qa4  
Bb6 14.Nbd2 Bb7 15.Ne4 Qf5 16.Bxd3 Qh5 17.Nf6+ gxf6 18.exf6  
Rg8 19.Rad1 Qxf3 20.Rxe7+ Nxe7 21.Qxd7+ Kxd7 22.Bf5+ Ke8  
23.Bd7+
```

27°C 9:49 PM 11/13/2024

Chessverse

127.0.0.1:5500/chess-console-master/examples/game-with-random1.html

## Chessverse

### Local vs Stockfish(Level 1)

Random Player

1. d4 d5  
2. e3 e6  
3. e4 h6  
4. c3 ♟f6

27°C 9:51 PM 11/13/2024



# CONCLUSION

Currently, the Chessverse project has a successful login and signup portal that leads to the main index page. We also have a working subscription option that leads to the payment gateway. Furthermore, we provide the users with different gameplay options that ensure that a novice as well as an expert can enjoy the game equally. We want to make sure that the user identification system works properly and set the foundation for the main gameplay mechanics.

We had previously set up the fundamental base of the game, which includes the game board and its components along with the movement of chess pieces according to the universal chess rules. With clarity in this region, as in a successful building of the foundation; we have now expanded to different gaming modes. A player can now play at a local level, that is with friends. While doing so, the game also records all previous moves that would later help the player understand or explain the movement.

Furthermore, a player can compete against a random bot which allows them to learn the mechanics of the game through trial and error. Once the player learns about these mechanics they can level up and compete against an AI (named Stockfish).

We now aim to build a system that tracks the user's overall progress that would in turn allow them to participate in online multiplayer games. These early achievements have made us aware of the challenges we might face while building a full-fledged online platform but we are excited and motivated to continue working on it and deliver an engaging chess platform.

In conclusion, this project has provided us with an incredible opportunity to sharpen our skills and also, learn new ones along the way. We look forward to bringing our vision to life.

# FUTURE SCOPE

For future development of the project, we plan on integrating multi-factor authentication processes in the form of password and OTP that adds the extra layer of security. This verifies the user and also maintains the player's progress. Furthermore, aspects like session expiry and renewal enhances security while maintaining user convenience.

Additionally, we intend to add a ranking algorithm that dynamically updates the player's rank after each match. Change of rank over time can show the player's advancement or regress.

We also plan on incorporating an Online Multiplayer server. This would require the implementation of an algorithm that finds opponents with similar skill level or ranks to ensure a fair match.

Lastly, we intend to add post-game statistics that can help the player discover and reflect on their blunders. Thos can significantly enhance the learning experience.

# REFERENCES

- Allen Newell, J.C. Shaw, H.A. Simon, "**Chess-Playing Programs and the Problem of Complexity**". Published in: IBM Journal of Research and Development, October 1958.
- Vikrant Chole, Vijay Gadicha, "**Hybrid Optimization for Developing Human Like Chess Playing System**", 2022 IEEE 3rd Global Conference for Advancement in Technology (GCAT).
- Jacek Mańdziuk, "**Towards Cognitively Plausible Game Playing Systems**", IEEE Computational Intelligence Magazine, 2011.
- Chatgpt.com
- <https://github.com/jhlywa/chess.js/blob/master/README.md>