SteemChess

Design Document
(Version 1.2)

Authors:

Warren Devonshire, Nisarg Brahmbhatt, Troy Pastirko, Matthew Halloran, Michael Naples, Daniel Haluszka, Alexander Boyle.

Distribution is subject to copyright.

Disclaimers :

Privacy Information :

Table of Contents :

6. Midterm Assessment Goals

1. Introduction

1.1 Purpose of This Document

The purpose of this document is to outline and provide a full understanding of the design of the SteemChess website and its integration with the Steem blockchain. Design choices, specifications, layout, and technology used will all be discussed in detail.

1.2 Scope

The purpose of our project is to create a chess game client and network on the Steem blockchain. The Chess game will integrate multiplayer capability through means of peer-to-peer connections. It will also include built-in social networking features, such as posting content related to chess and reacting to content posted by other users. All transactions within the game and the social network will be hosted on the Steem blockchain. Because of the integration with the Steem blockchain, all users who actively produce content within the game network will be able to earn revenue for that content in the form of Steem cryptocurrency. More information about the Steem blockchain can be found at: https://steem.com/

2. Design

2.1 Constraints

SteemChess will use a blockchain for many of its transactions, which limits the rate at which transactions can be processed due to the way that they are verified. This also prevents SteemChess from fully controlling or guaranteeing the reliability of the network. In addition, SteemChess cannot guarantee that all users will earn revenue for the content that they create due to the method used to distribute Steem cryptocurrency. Further detail about this method can be found on the Steem website.

2.2 Solution Design Overview

The two most viable approaches for the backend of the SteemChess website were either utilizing the Steem blockchain, or taking a more traditional server approach for handling transactions. Between these two options, the blockchain was chosen for a variety of reasons.

One reason that the blockchain was selected was because of the transparency and immutability that it provides; More specifically, all users of the blockchain have unlimited access to the same data. This data, once verified through means of consensus, becomes immutable on the blockchain.

This immutable nature allows for high traceability of all data posted. Anyone can fetch data from the blockchain at any time, a property which lends itself well to a chess application in which the results of matches must be verified. Having almost all essential data on the blockchain also simplifies what the SteemChess website needs to host and store in order to operate.

In addition to all of the benefits of using the blockchain described above, there were also reasons that made the server approach less desirable. Administration in a server approach is handled by a handful of people, which can limit access to data, as well as lead to compromises in this data if a security breach were to occur. In addition, the issue of server upkeep is placed entirely on the SteemChess website, leading to costs which aren't incurred when using a blockchain approach.

3. Specifications

3.1 Performance Requirements

For achieving optimal performance, any computer with minimum specifications for everyday use and a reasonably fast internet connection will be sufficient to utilize the full functionality of the SteemChess website.
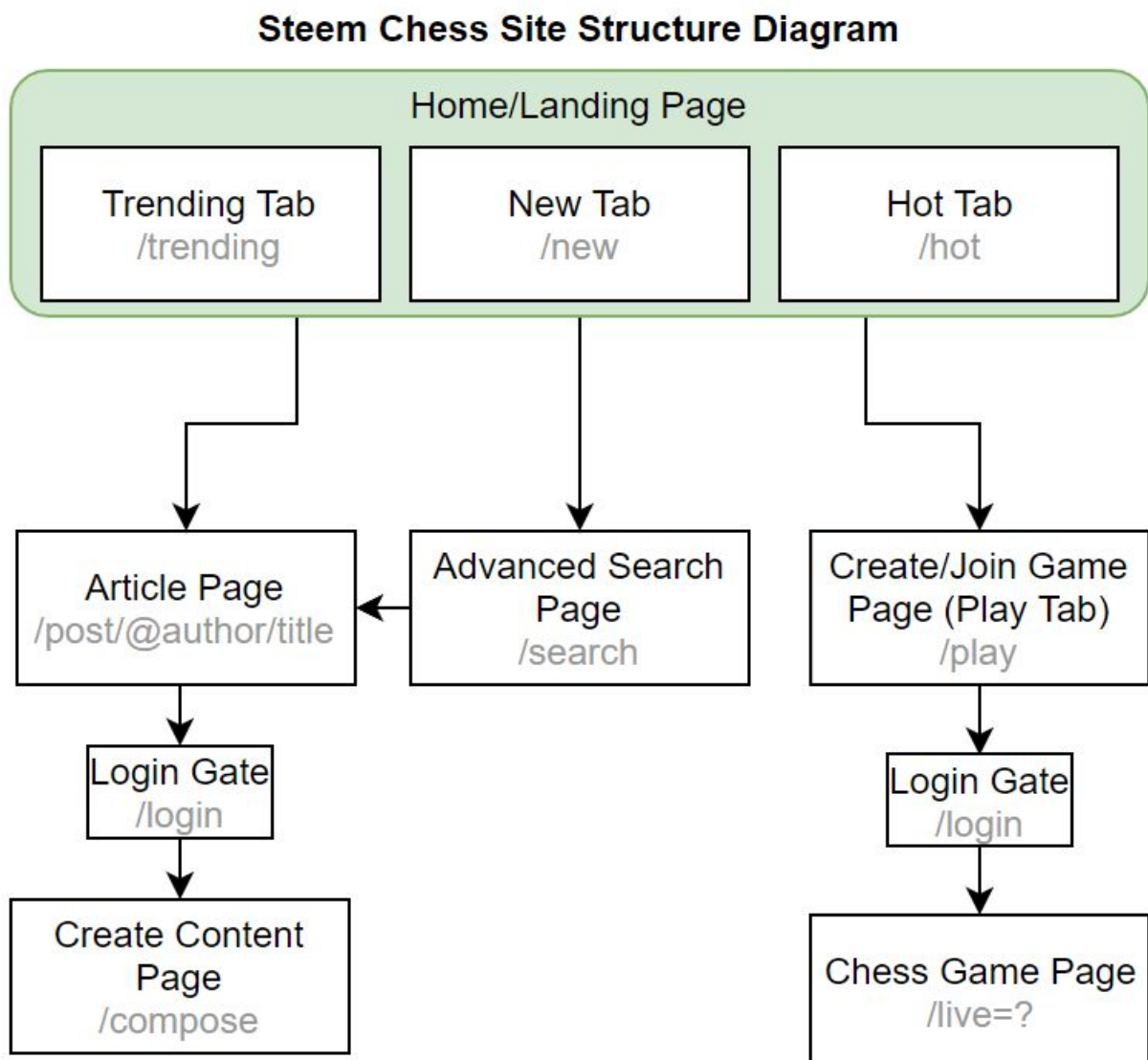
3.2 Hardware Requirements

Being a web-based application, there are no additional hardware requirements, as long as the user has a Computer / Laptop / IPad / Tablet.

4. Layout

4.1 Site Structure Diagram

This diagram outlines the pages on the SteemChess website and how they associate with each other.

Steem Chess Site Structure Diagram

Home/Landing Page

Trending Tab /trending	New Tab /new	Hot Tab /hot

Article Page /post/@author/title ← Advanced Search Page /search

Create/Join Game Page (Play Tab) /play

Login Gate /login

Create Content Page /compose

Login Gate /login

Chess Game Page /live=?

4.2 Home Page
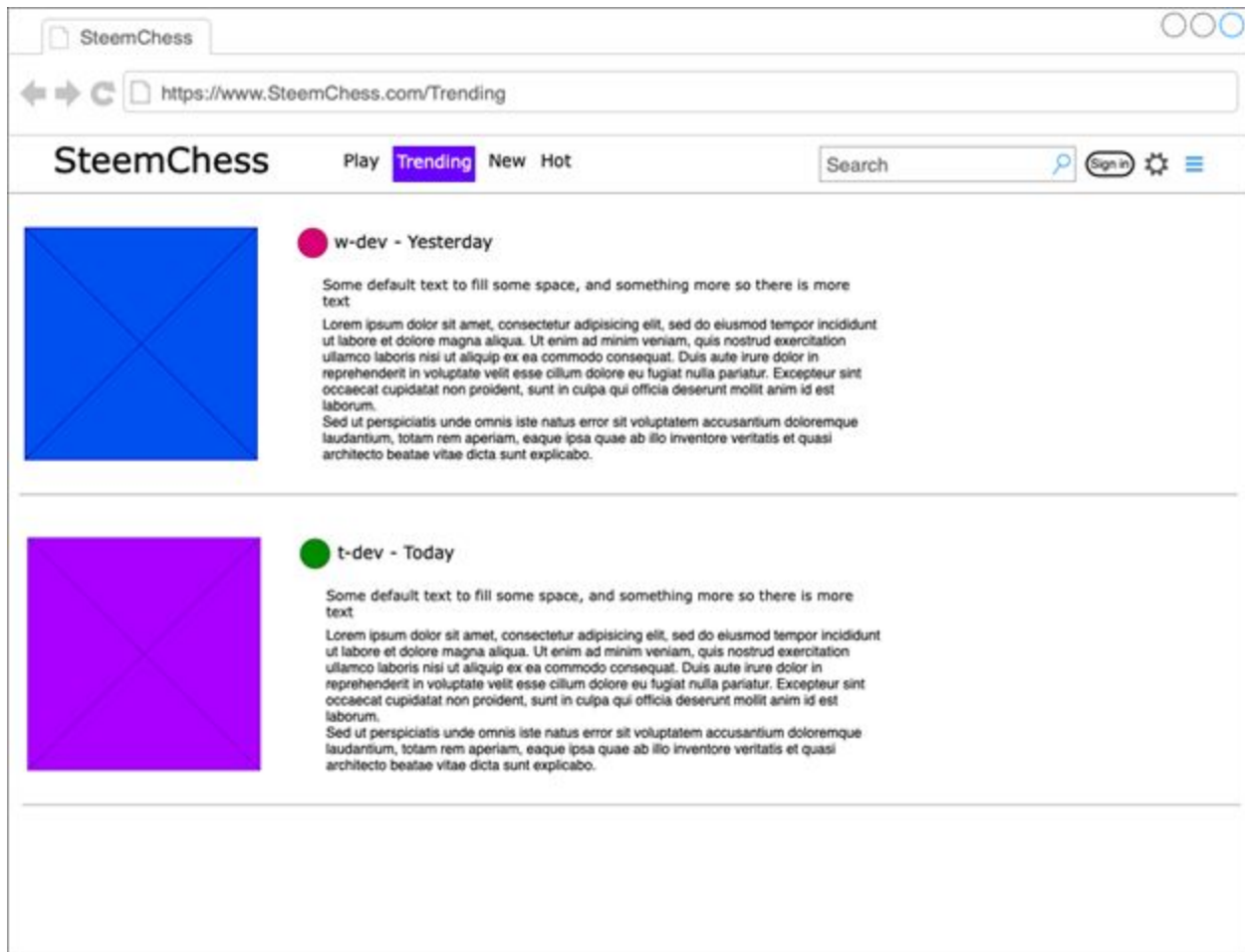
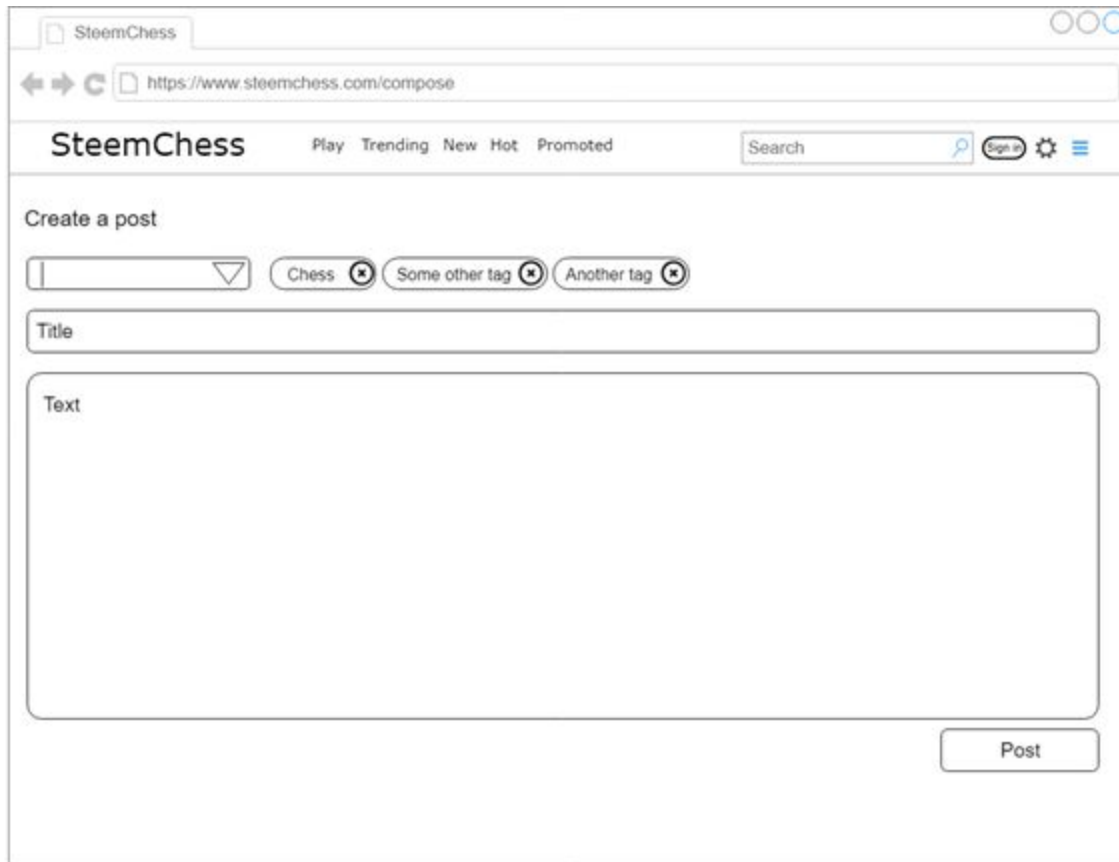This is the primary screen from which all options for user functions will be available. The functions available are as follows: 'Play Game' which leads to the Play Game Page from which the user can create or join a game. The 'Trending,' 'New,' and 'Hot' tabs at the top of the page will all change which content is displayed in the center of the page and how it is sorted. The search bar will allow a user to type in a term or multiple terms that they wish to search all chess content in the network for. Users will be able to sign in through Steem when they click the 'Sign In' button. Finally, content displayed in the center of the page will allow users to view the full body of the content when clicked on. All functions in the main menu bar will carry over to every other page on the SteemChess website, as the menu bar is persistent.

4.3 Article Page

        This page will display the full body of a piece of content when that content is clicked on. Users will be also be able to view the tags associated with this particular piece of content, as well as all comments on it. If the user is logged in, they will also be able to post their own comment on the article.

4.4 Compose Page

      This page will only be accessible after passing through a login gate. Once a user is logged in, this page will allow them to create a new post on the SteemChess network which will be posted to the Steem blockchain. The user will be able to choose tags to associate their post with, choose a title for their post, and compose the body of the post. Once complete, their content can be posted using the post button. All new posts created through the SteemChess website will be required to have the tag "Chess" associated with them.

4.5 Play Game Page

Create Game (Real Time)

SteemChess — Play | Trending | New | Hot | Search | Sign in

Create Game

- ● Real Time
- ○ Correspondence

Time Per Side — 10 minutes

Increment — 10 seconds

Starting Color

Start

Join Game

Filter [Most Recent ▽]   View ○——

Name	Type	Time	Posted
John Smith	Real Time	10-0	2 Minutes
John Smith	Real Time	10-0	2 Minutes
John Smith	Real Time	10-0	2 Minutes
John Smith	Real Time	10-0	2 Minutes
John Smith	Real Time	10-0	2 Minutes
John Smith	Real Time	10-0	2 Minutes
John Smith	Real Time	10-0	2 Minutes
John Smith	Real Time	10-0	2 Minutes
John Smith	Real Time	10-0	2 Minutes
John Smith	Real Time	10-0	2 Minutes
John Smith	Real Time	10-0	2 Minutes
John Smith	Real Time	10-0	2 Minutes
John Smith	Real Time	10-0	2 Minutes
John Smith	Real Time	10-0	2 Minutes
John Smith	Real Time	10-0	2 Minutes

Join

Create Game (Correspondence)

SteemChess — Play | Trending | New | Hot | Search | Sign in

Create Game

- ○ Real Time
- ● Correspondence

Days — 5 Days

Starting Color

Start

Join Game

Filter [Most Recent ▽]   View ——○

| 3 Days	1 Day	2 Days
John Smith 2 minutes	Lebron James 1 minute	James Cameron 5 minutes
1 Day	3 Days	1 Day
Lebron James 1 minute	John Smith 2 minutes	Lebron James 1 minute
3 Days	3 Days	2 Days
John Smith	John Smith	James Cameron

Join

*Join game pictures will be a chess board,
oriented to show what color they chose

Page 9

This page will only be accessible after passing through a login gate. This page is the landing page for clicking the 'Play' tab on the menu bar and will allow users to create or join a chess match. The menu for creating a new match is on the left side of the screen, while the menu for joining an existing match is on the right side. When creating a new match, the user will be able to select the type of match they want to play, the time limit for moves in correspondence games or the match in real-time games, and the color they'd like to start as. When joining an existing match, the user will be able to view a list of matches that are waiting for a player, and select which one they'd like to join based on different filters.
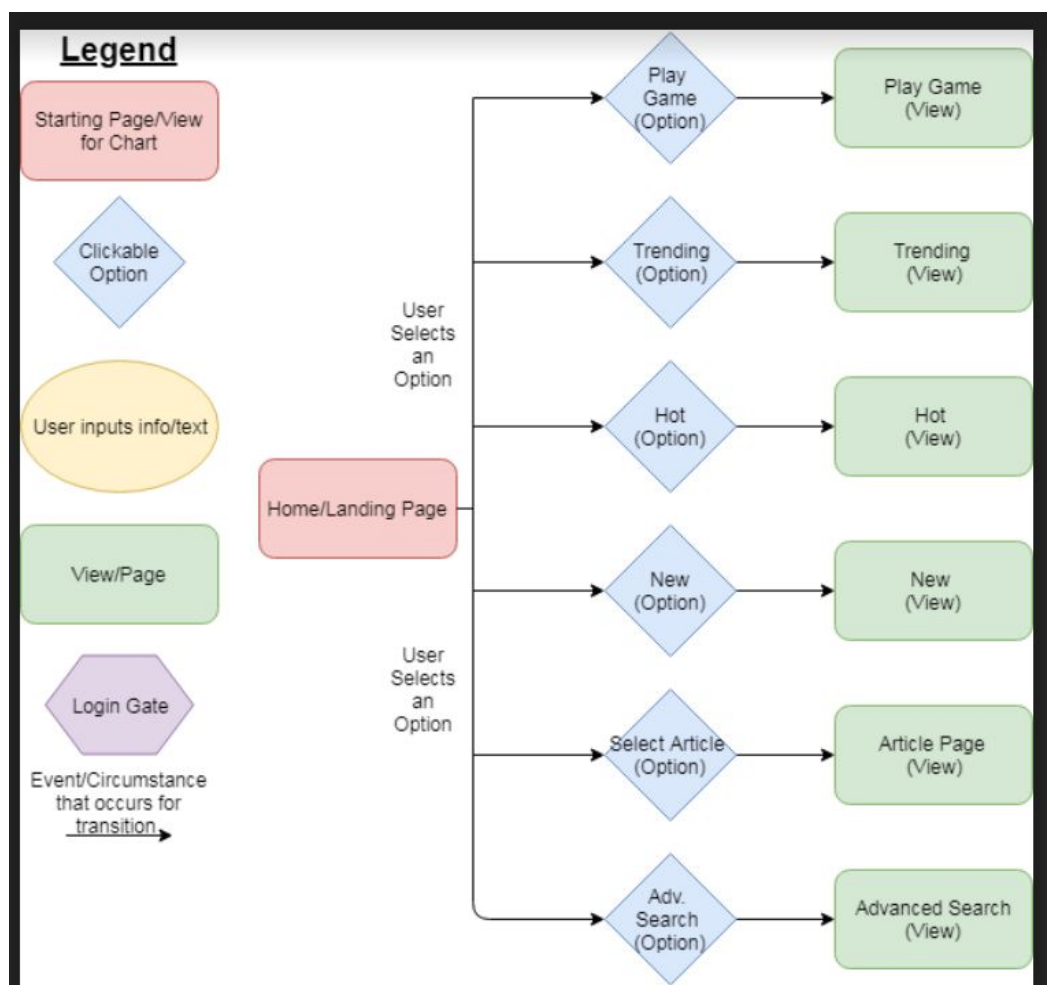
4.6 Live Game Page
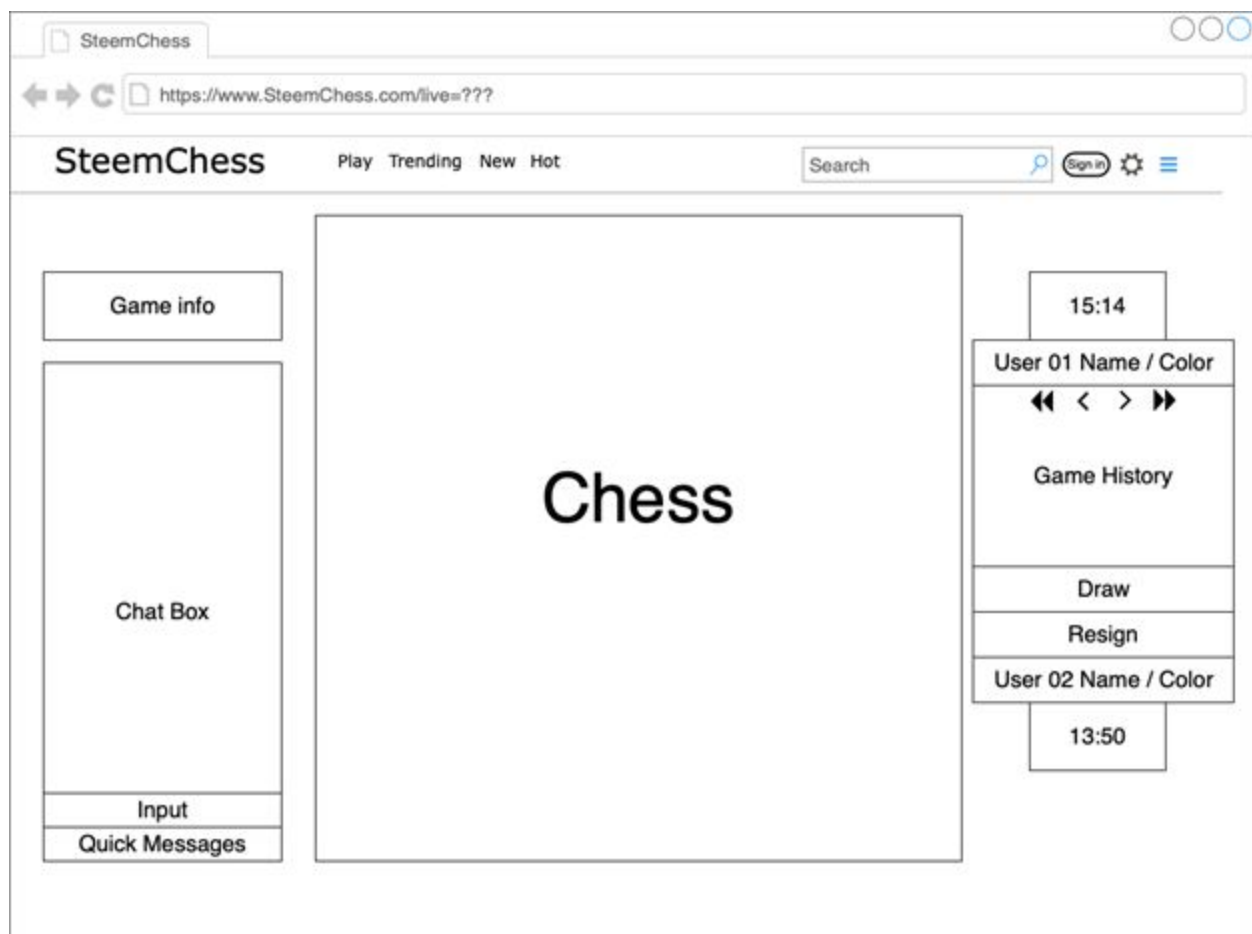
This page will only be accessible after passing through a login gate. This page will display the current game of chess that the user is playing. Information about the match as well as the in-game chat between the two players is on the left side. The chess board is in the center of the screen, and will allow the user to make moves through an intuitive user interface. On the right side of the screen, the user will be able to view the time remaining in their turn or the match, view who's turn it currently is, view move history for this match, and offer a draw or resign the current match.
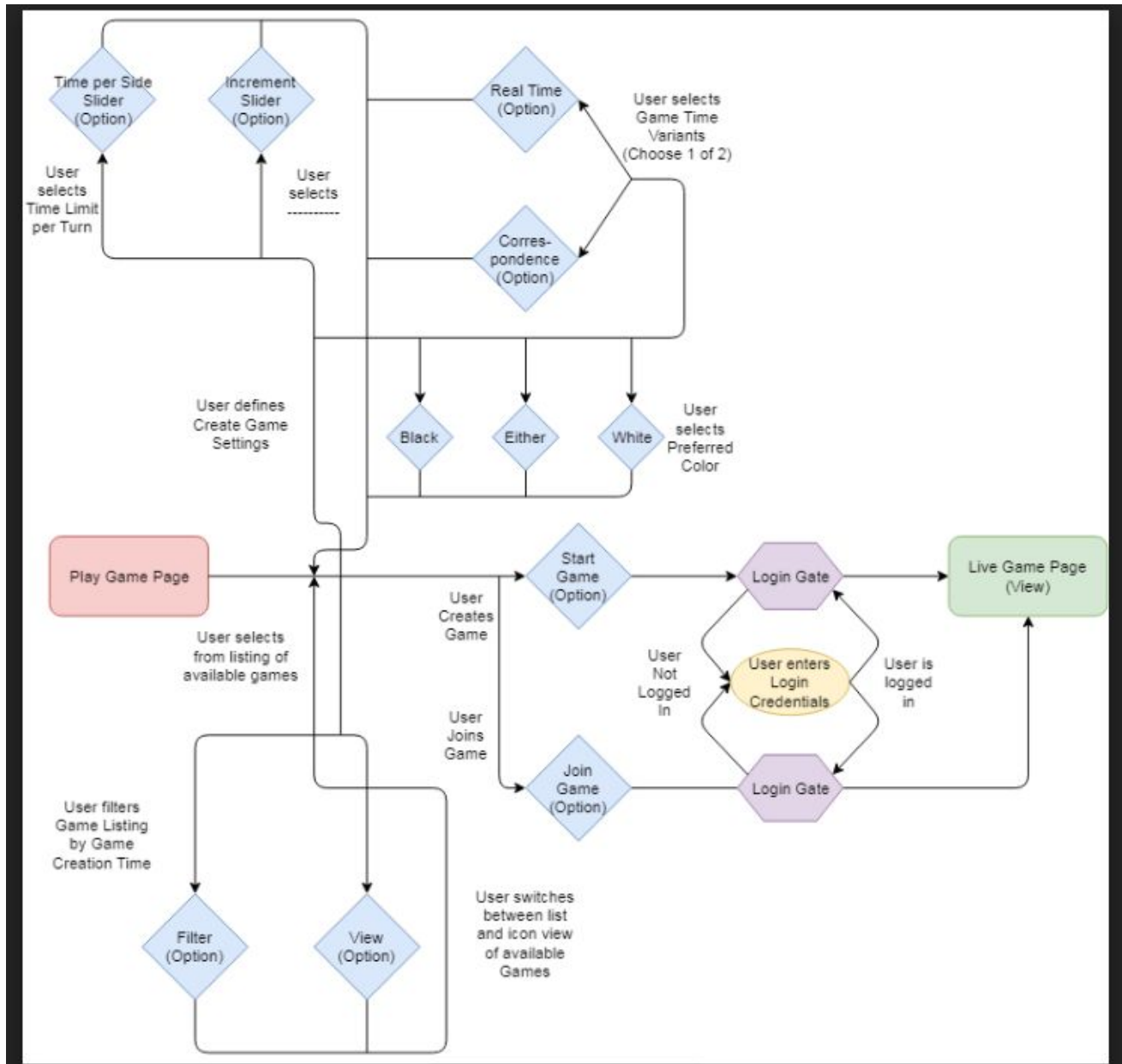
4.7 Flow Diagrams

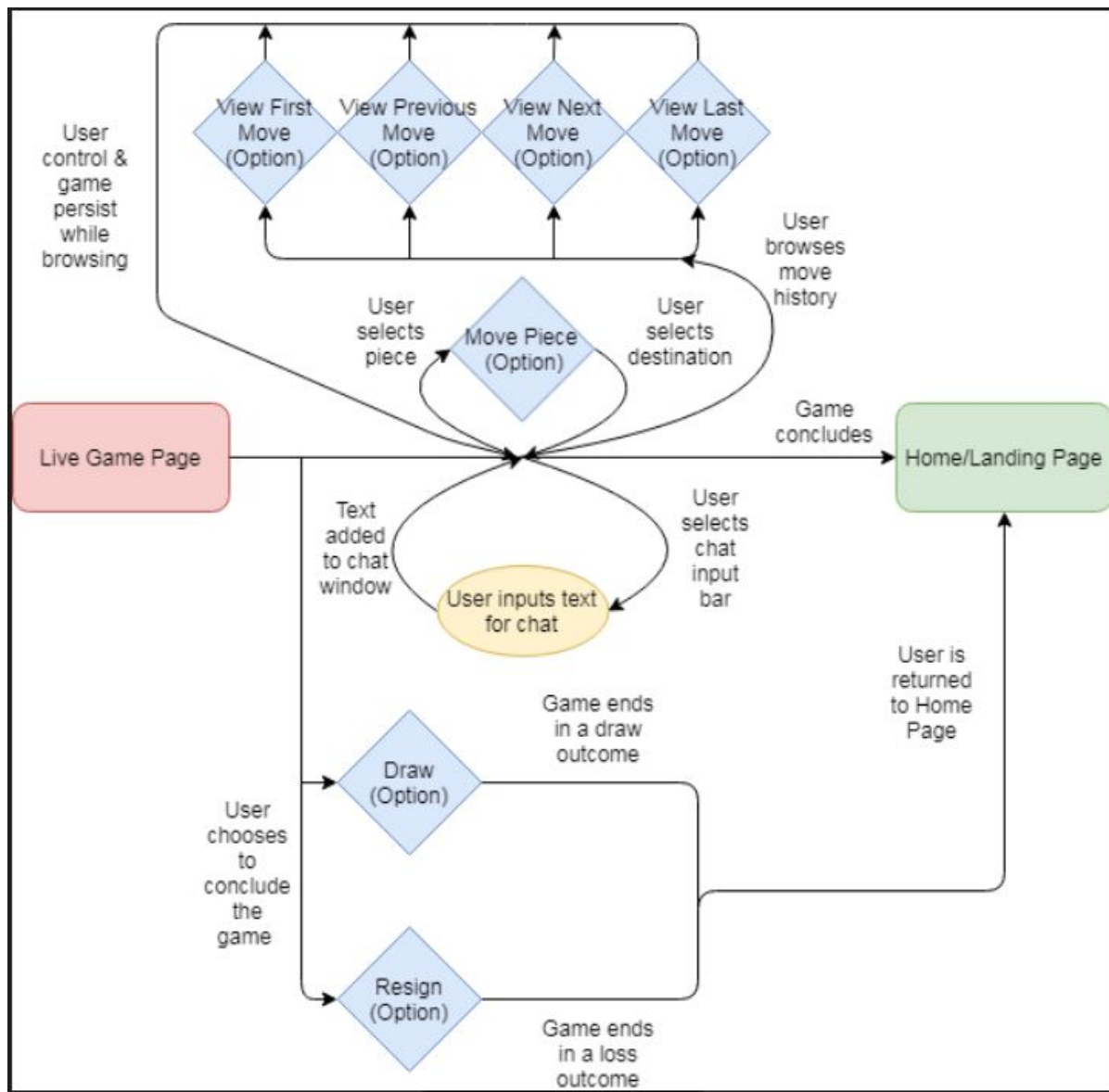The flow diagrams outline user paths to various actions on the website.

1. Home Page Flow Diagram

2. Play Game Page Flow Diagram

3. Live Game Page Flow Diagram

5. Technology

5.1 Technology Stack

The frontend of the website is being developed in HTML, CSS3, JavaScript, react.js, and the chessboard.js library. The backend of the website is being developed using node.js, dsteem.js, and the chess.js library. Since the website involves in-game communication, we'll also be using WebRTC.

React.js will be our frontend framework for building user interfaces. Our frontend also contains CSS3 for web page styling, and Javascript for scripting purposes. The chessboard.js library will be used to create and update the GUI for playing a game of chess.

For our backend, we have node.js, which we will be using for event-driven input/output purposes. With the help of WebRTC framework, we will be enabling peer-to-peer communication in the browser. We will be using the dsteem.js library for both pulling blogs related to chess from the blockchain and to enable matchmaking. The chess.js library will be used to handle the backend data and logic involved in a game of chess.

5.2 Backend Information

The website itself will be hosted on Amazon Web Services (AWS). Node.js will serve to enable the use JavaScript to create and handle user actions on the site. Access to the Steem blockchain will be provided by the dsteem.js library. The Steem blockchain will be used to handle most transactions on the site. These transactions include: Connection negotiation, saving and broadcasting game results, broadcasting match move data for correspondence games, as well as fetching, posting, and reacting to articles or other content. For real-time matches, move data will be sent via the peer-to-peer connection that will be established between the two users in the match. This peer-to-peer connection, which will be set up via the Steem blockchain and WebRTC at the beginning of a match, will also serve as the backend for the ingame live chat for real-time matches. Finally, the chess.js library will be used to implement all of the logic and move validation needed to play a game of chess.

5.3 Authentication and Security

Authorization on the SteemChess website will be done using the same username that a user inputs for SteemIt, along with their private posting key. Login information will only be stored via the local storage of the browser.

5.4 Input-Output

Input: The primary method of user input will be using buttons to navigate each page of the website as well as making moves in a chess game. The website will also receive text inputs when the user is typing a message in the chat box for communication with their peers.

Output: The output provided to the user will be text and images in the content that they view. Additionally, output when playing a game of chess will include an intuitive view of the current game and the information regarding it.

6. Midterm Assessment Goals

- Login with SteemIt credentials
- Post content to the blockchain.
- Post comments to the blockchain.
- Upvote content.
- Play a peer-to-peer Chess game.
    - Establish a WebRTC connection.
    - Negotiate a peer-to-peer connection through blockchain.