Birla Institute of Technology

Software Engineering Lab

Topic:Chess Masters Club

Team Members:
Rohit Kumar (BE/10035/17)
Rahul Agarwal (BE/10047/17)
Shitanshu Bhushan (BE/10076/17)
Madhav Kauntia (BE/10147/17)

ACKNOWLEDGEMENT

We owe a great many thanks to a great many people who helped and supported us during our project work, especially to Prof. Bhaskar Karn for this wonderful project which is not just confined to textbook knowledge but has a real-world application and helped us gain significant practical knowledge which will be necessary while pursuing our respective careers. He has taken pain to help and guide us during these difficult times. We also convey our regards to all other faculty members of the Department of Computer Science and Engineering, Birla Institute of Technology, Mesra. Finally we would like to thank all our family and friends for their assistance and motivation throughout this project.

ABSTRACT

Birla Institute of Technology, Mesra is one amongst the most reputed institutions for technical education in India. The main purpose of the project is intended to develop a portal for playing multiplayer chess online. This portal provides a convenient and an easy to use User Interface which can help its users to play chess online in real time with their friends. This portal was mainly developed using various technologies like HTML, CSS, Javascript, NodeJS, MongoDB and Socket.IO .

CONTENTS

1. INTRODUCTION

1.1 NAME OF THE PROJECT:

Chess master club

1.2 PURPOSE:

The purpose of this document is to build a web-based chess game that can be used by chess loving people to interact and play games.

1.3 INTENDED AUDIENCE/USERS:

The intended audience is as follows-

A. General end user for fun
B. Chess tournament players
C. Admins/managers

1.4 PROJECT SCOPE:

Creating and managing an online chess club ,where game lovers can learn and play chess games by different means (chess tutorial, game with computer or other players etc.) . Professionals can take part in tournaments that are totally a commercial activity for site owners.

2. REQUIREMENT SPECIFICATION

The purpose of SRS (Software Requirement Specification) document is to describe the external behavior of the web based chess game. It defines the operations, performance and interfaces and quality assurance requirements of the web based chess game. The complete software requirements for the system are captured by the SRS (Software Requirement Specification).

2.1 FUNCTIONAL REQUIREMENT:

For documenting the functional requirements, the set of functionalities supported by the system are to be specified. A function can be specified by identifying the state at which the data is to be input to the system, its input data domain, the output data domain, and the type of processing to be carried on the input data to obtain the output data.
Basically the management parts are the functional requirements which are as follows:-

1. Registration and profile management for players with their preferences Players can choose from beginner and expert level while registering. Initially the player will be ranked as a beginner.
2. Provision for chess pieces movement policies, players cannot violate the rule of the chess game and have to play according to the rule.
3. Player rating evaluation algorithm, players will be rated according to their performance in the game . It will be decided based on the number of wins,losses and draws played by the player.
4. Tournament game management activities, tournaments will be organised on the monthly basis , where only trusted players will be allowed to take part in their respective divisions, It will be like a knock out round where each player will be divided on the basis of their rank.
5. Game stop,resume and saving options,players will be able to stop their game and resume it again.
6. Help tutorial and instructions for end users to learn chess game, a documentation will be provided to the players so that they can learn chess, they can learn new moves, and rule and regulations of the game.
7. Strategic data and graphs for admin and authority, so that the admin can implement the rating algorithm.
8. Basic and advance admin facilities like backup/recovery of data, Generating various reports.

2.2 NON-FUNCTIONAL REQUIREMENT

These are the requirements that are not functional in nature. Especially these are the constraints the system must work within.

Performance Requirements: The system response time must be less than 30 seconds for the user interface. Or else the system will show TIMED OUT, Board will be loaded once in the browser then we send geometric notation of the board for refreshing ,so the board will be refreshed in microseconds.

Reliability Requirements: The system shall have a minimum uptime of 99 % excluding time pre-scheduled for maintenance and/or upgrades.

Safety Requirements: All the system data must be backed up every day and the backup copies stored in another server at different locations for disaster recovery,use of ssl for data transmission during game.

Quality Attributes: The source code for the system is well documented for ease of maintenance and upgrading the system in future.

Availability : 24x7

3. DESIGN

3.1 Data Flow Diagram

Data Flow diagram is a graphical representation of flow of data throughout the information system. Data flow diagrams illustrate how data is processed by a system in terms of inputs and outputs.

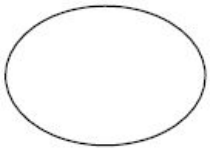
Name	Notation	Role
Process		Transforms incoming data flow to output data flow
Data Store		Repositories of data in the system.
Dataflow		Dataflow are pipelines through which packets of information flow.
External Entity		External entities are objects outside the system, with which the system communicates

Table 1 - Data Flow Notation

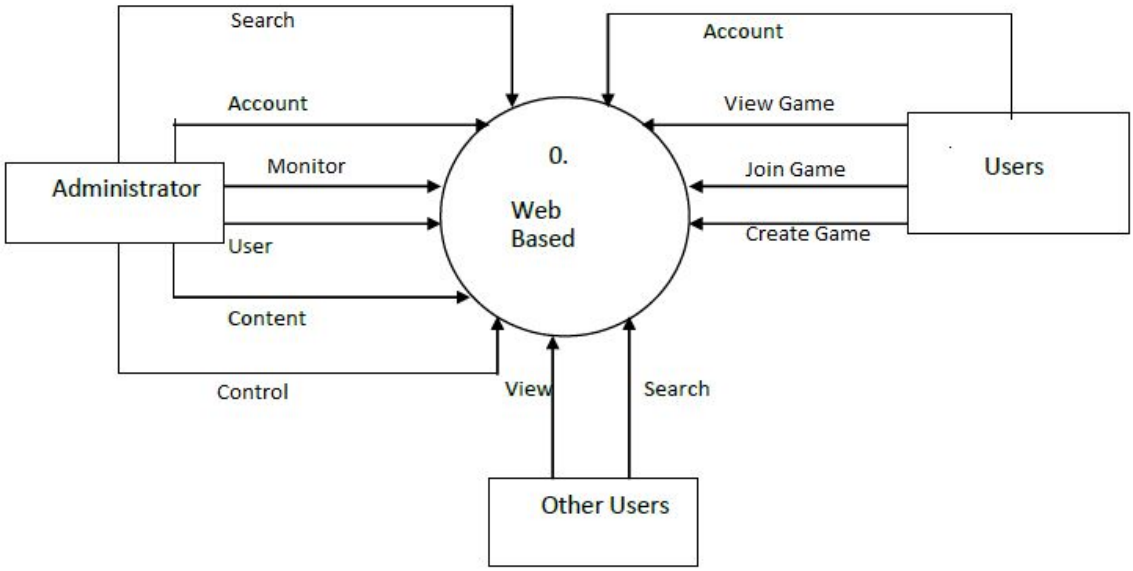
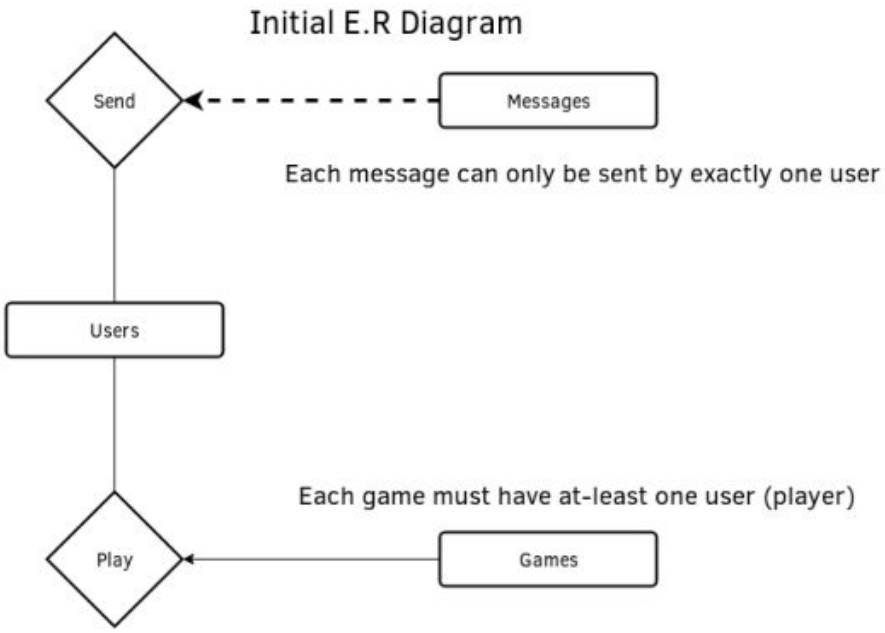
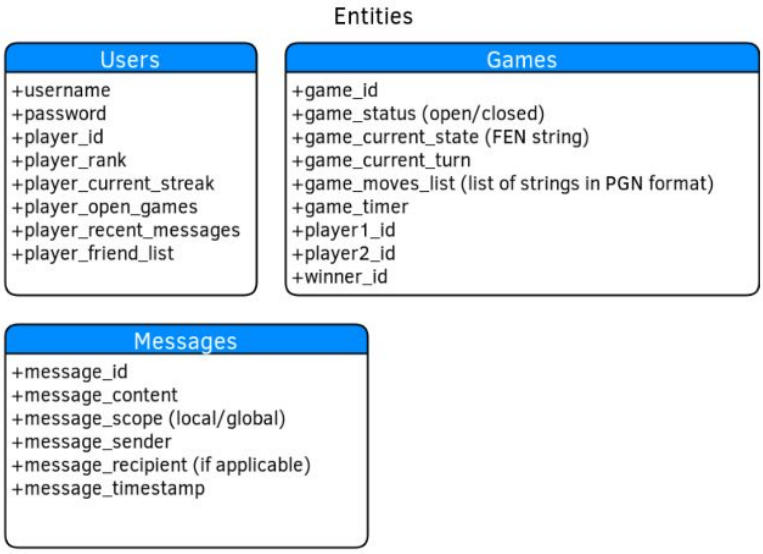


FIGURE 1 – CONTEXT DIAGRAM

3.2 Entity Relationship Diagram

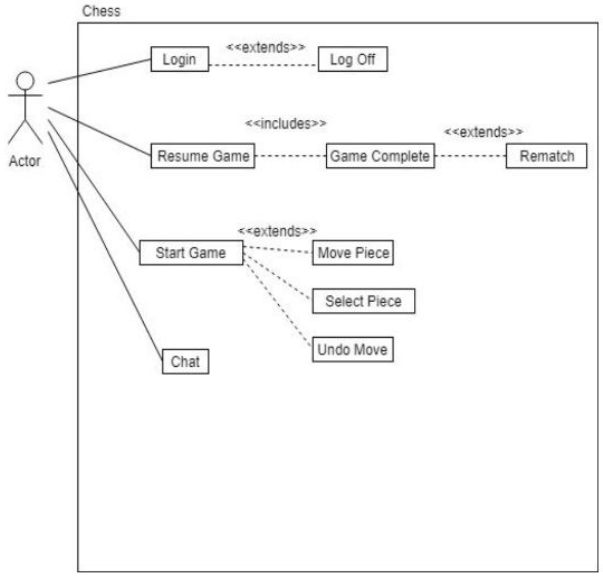


3.3 Database Design



3.4 Use Case Diagram

Use Cases:



4. IMPLEMENTATION

Implementation or execution in any project is done when the project planning is completed and is as detailed as possible. In this phase, we monitor everything that could jeopardize the project or part of it. In other words the theory is being tested by implementation to make sure the data in the system changes according to plan. In order to keep track of project plans and keep ourselves in control all the tools and steps involved are applied by the project manager. Steps in which all the tools are used:

I. Take action according to plan
II. Record and document all the work, changes & results
III. Analyze the result of action or changes made according to the plan

The steps involved here in this particular phase are:

1) Installation of Hardware and Software
2) Conversion of data
3) System Training
4) Documentation of the system

Installation of hardware and software: Before implementation the hardware and the software required for debugging the system should be made fully operative so the translation can occur smoothly which is one of the most critical and costly activities in the system development life cycle.

Conversion of data: As the old system is converted into a new system similarly the data from the old system should also be converted to run in the new system within the new format. In this step the recovery procedures are needed to be fully defined and also the database is configured with security. The data format we used is the NoSQL format as opposed to the table format in SQL database.

System training: All the programs of the system are loaded onto the computer systems of the user during this step. After loading the system we check by whether the project is under the control of the user and works according to the plan.

To confirm the above sentence we need to complete the following steps:

I. To execute the packages used in our project.
II. To pass the data from one page to another
III. To process the data in our databases via the mongo server
IV. To take out the reports/records and test our results

As the new system is completely trained about the computerized system then the working is also changed from manual to computerized working and the process is called “Changeover”.

Documentation of the system: This is also one of the main activities in the system development life cycle. This ensures the stability and continuity of the system. There are basically two types of documentation. These are:

I. **User or Operator Documentation**: It is a complete explanation of the system from the end user’s point of view how to use or operate the system. It contains the major error messages likely to be run into by the users. For Example : not to register again if already registered etc.

II. **System Documentation**: It comprises the details of system design, programs, coding, system flow, data dictionary, etc. This helps to realize the system and permit changes to be made in the existing system to fulfill new user needs.

4.1 TECHNOLOGIES USED

1) NodeJS

NodeJS is a free, open sourced, cross-platform JavaScript run-time environment that lets developers write command line tools and server-side scripts outside of the browser. We used nodeJS extensively in the backend of our project. Some of them are:

I. Almost all the functions related to the server side such as login, etc.
II. Running scripts in the frontend

2) MongoDB

MongoDB is a cross-platform document-oriented database program. Classified as a NoSQL database program, MongoDB uses JSON-like documents with optional schemas. We used it to create two databases, namely User and Articles to store user information and the articles which are posted by the users. All of the functions related to the database such as posting, deleting, updating or reading items in the database were managed by a mongo server.

3) Socket.IO

Socket.IO enables real-time bidirectional event-based communication. It consists of:

- a Node.js server
- a Javascript client library for the browser (or a Node.js client)

It works on every platform, browser or device, focusing equally on reliability and speed. Socket.IO is built on top of the WebSockets API (Client side) and Node.js.

Its main features are:

Reliability
Connections are established even in the presence of:
- proxies and load balancers.
- personal firewall and antivirus software.

Auto-reconnection support
Unless instructed otherwise a disconnected client will try to reconnect forever, until the server is available again.

Disconnection detection
A heartbeat mechanism is implemented at the Engine.IO level, allowing both the server and the client to know when the other one is not responding anymore.

Binary support
Any serializable data structures can be emitted, including:
- ArrayBuffer and Blob in the browser
- ArrayBuffer and Buffer in Node.js

4) HTML and CSS

In this project HTML, CSS codes are used.

HTML (Hypertext Markup Language): It is the main markup language for displaying web Pages and other information that can be displayed in a web browser. HTML written in the form of the HTML elements consisting of tags enclosed in angle brackets within the web page content. The purpose of the web browser is to read HTML documents and

compose them into visible web pages.The browser never displays the HTML tags these tags are used to interpret the content of the pages.

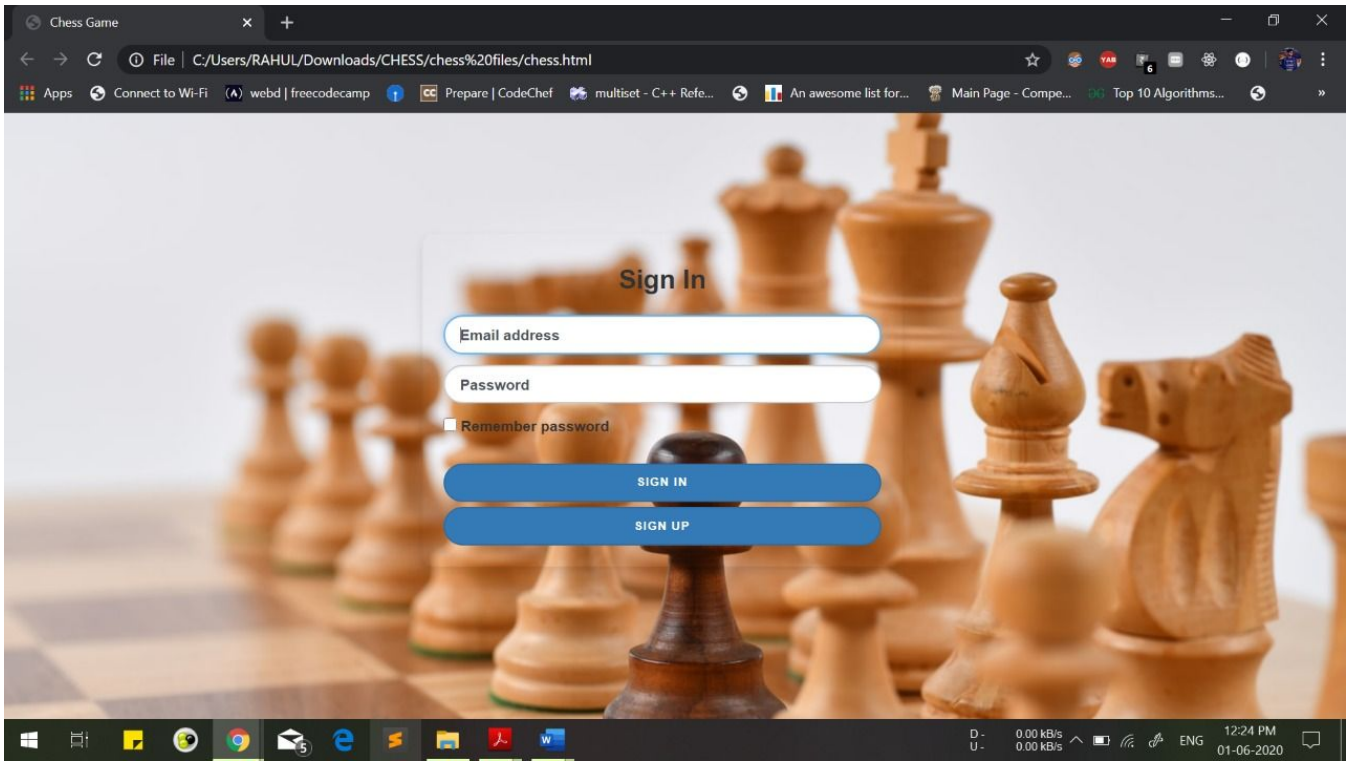
CSS (Cascading Style Sheets): It is referred to define the appearance and layout of text and other material. CSS is encouraged to be used over explicitly presentational HTML markup.

4.2 SCREENSHOTS AND RESULTS

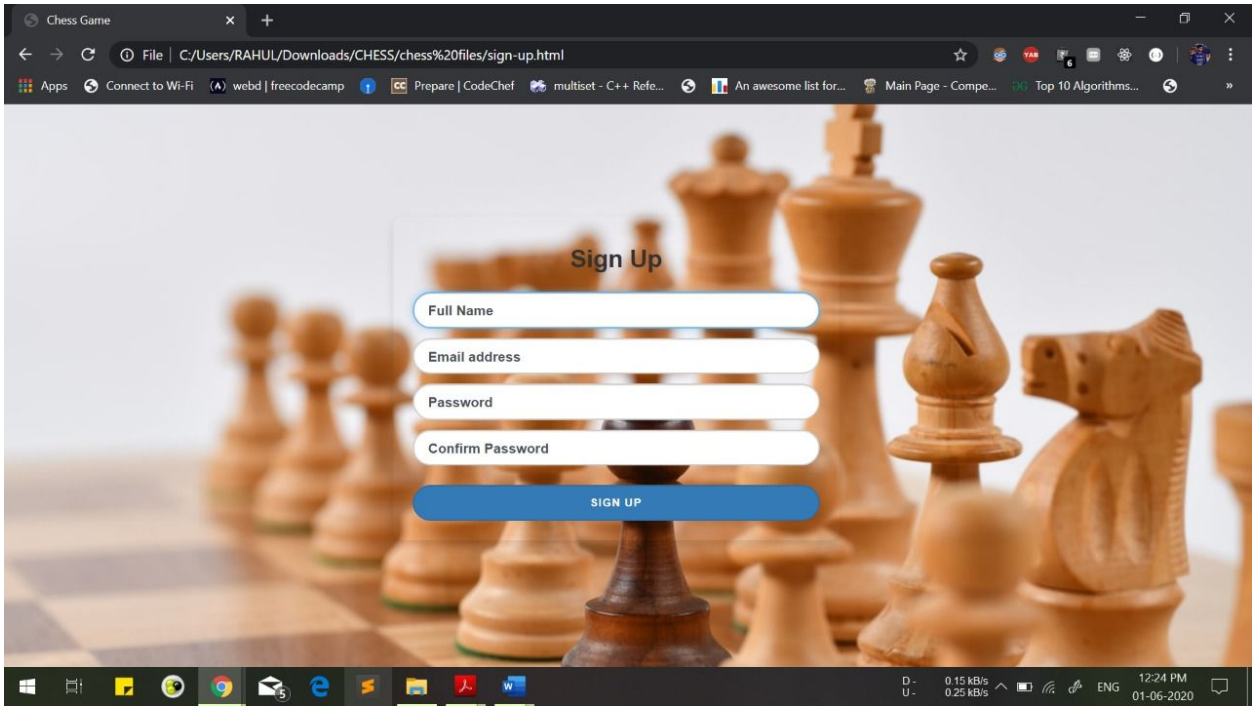
After going through all the phases of the System Development Life Cycle of this project, the portal is designed successfully.

Below is a glimpse of the various features of the website.

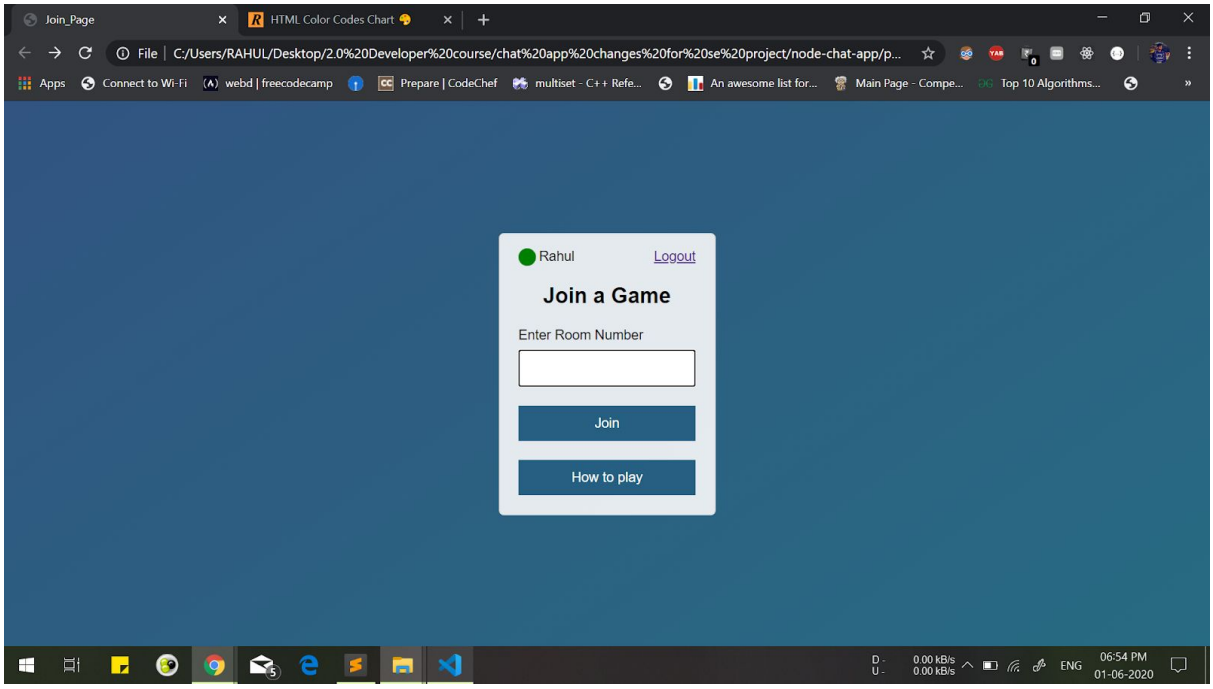
Screenshot 1: LOGIN PAGE - To login to the website using username and password.



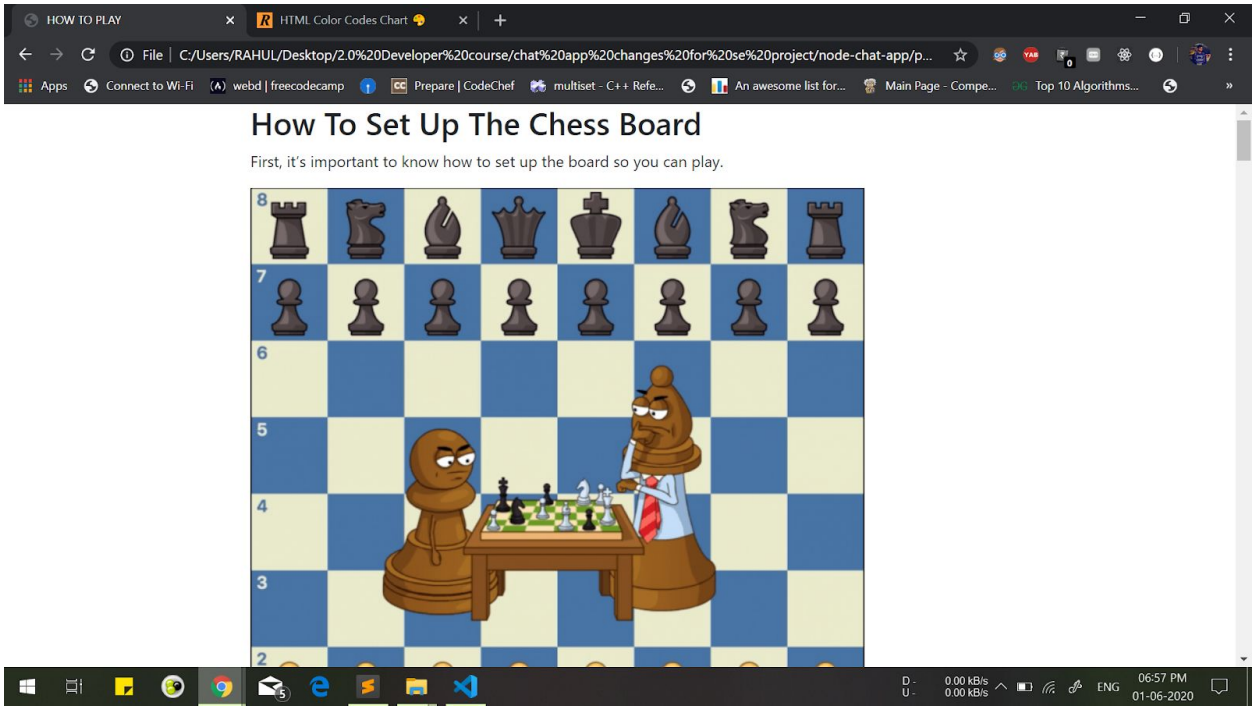
Screenshot 2: REGISTER PAGE - New user will have to register first using their Email id.



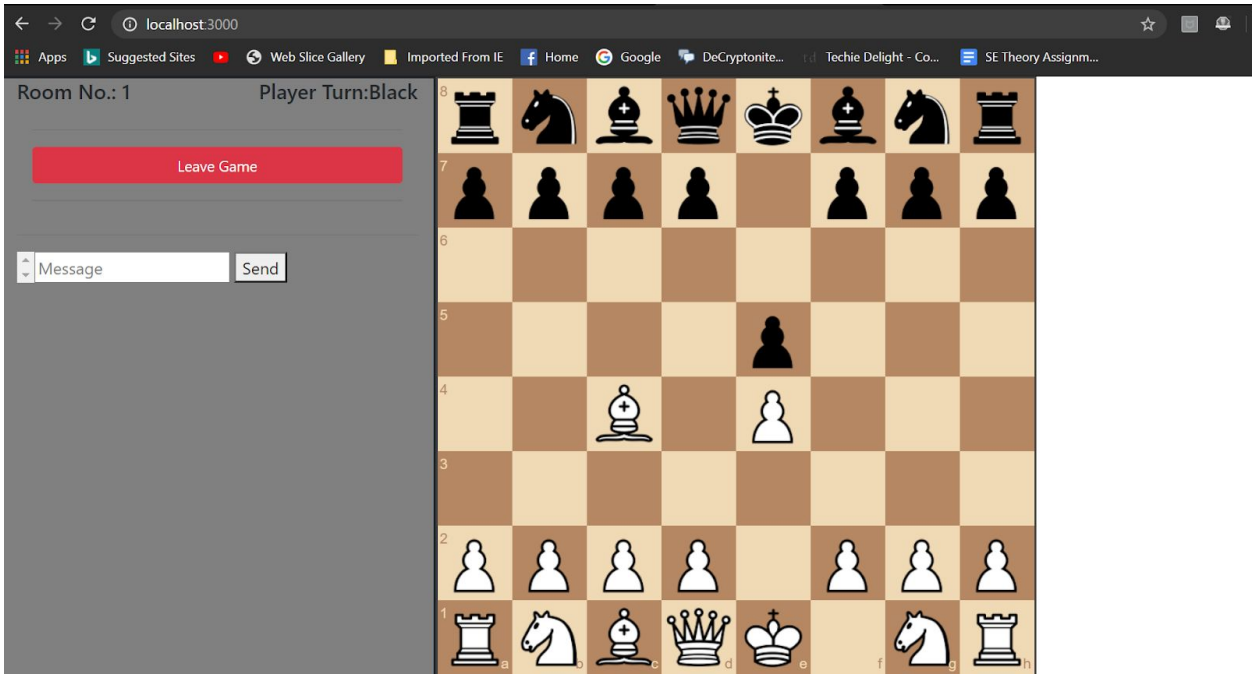
Screenshot 3: HOME PAGE - This is the main landing page of the website and from here a user can either start a game or learn how to play. To start a game , the user has to enter a room code to create a game and the opponent can enter the same number to join the game.



Screenshot 4: How-to Play PAGE - This page has information on how chess is to be played and its rules.



Screenshot 5: GAME PAGE - This is the page where the game will be played and also has a chat feature so that the two players can chat with each other while playing.



5. TESTING

Testing phase is very important for ensuring the system is built in the desired manner. This phase before implementing the new system into operation is essential, for eliminating bugs and having a test run of the system before deploying. After completing coding the whole system, a test plan should be developed and run one given set of test data.

Using the test data subsequent test runs are carried out:

Unit testing: It is a method by which individual units of source code, sets of one or more program modules collectively with associated control data, usage procedures, and operating procedures, are tested to determine whether they are fit for use. Intuitively, one can view a unit as the smallest testable part of an application. In procedural programming a unit can be an entire module but is more commonly an individual function. In object-oriented programming a unit is an entire interface but could be an individual method. Unit tests are created by programmers or by white box testers during the development process.

Each test case is independent from the others: substitutes like method stubs, mock objects can be used to assist testing a module in isolation. Unit tests are typically written and run by software developers to ensure that code meets its design and behaves as intended.

Integration testing: It is the phase in software testing in which individual software modules are combined and tested as a group. It occurs after unit testing and before validation testing. Integration testing takes as its input modules that have been unit tested, groups them in larger aggregates, applies tests defined in an integration test plan to those aggregates, and delivers as its output the integrated system ready for system testing .

Program test: After the programs have been coded, compiled and carried out to working conditions, they must be independently tested with the prepared test data. Any unwanted happening should be noted and debugged.

System Test: When the program test for each of the programs of the system is written and errors are removed then system test is complete. At this stage the test is done on actual data. The complete system is put into execution on the actual data. At every stage of the execution, the output of the system is studied. During the outcome analysis, it may be found that the outputs are not matching the estimated output of the system. In such situations, the bugs or errors in the particular programs are recognized and are fixed and further verified for the expected output. When it is confirmed that the system is running error-free, the users are called with their own real data so that the system could be presented running as per their requirements.

6. CONCLUSION

The chess game was developed after thoroughly processing through all the phases of the system development life cycle. In future it may be hosted on the internet, allowing its users to get to play chess games from all over the world. The Administrator or manager who will be assigned for managing will be given the secure login information and will have the authority to change or modify the website as per the requirements. All in all this project was a great experience for each member of our group and helped us gain a lot of insight about the process of ***Software Engineering*** and ***Software Development***.