Software Requirements Specification

1 Requirements Introduction

The San Jose Earthquakes Football Prediction Discord Bot aims to enhance the user experience for fans of the San Jose Earthquakes by providing a platform for predicting scores of football games. This unique bot allows users to submit their score predictions and earn points based on the accuracy of their forecasts compared to the actual game scores. The following document outlines the functional and performance requirements of this Discord bot.

1.1 Scope

The scope of the San Jose Earthquakes Football Prediction Discord Bot includes features such as score prediction submission, scoring calculation, leaderboard display, and user engagement through a user-friendly interface.

1.2 Referenced Documents

- Discord API Documentation
- Discord JS Documentation
- San Jose Earthquakes Football Game Data API

1.3 Requirements

The requirements section specifies various aspects of the bot, including its functionalities, user interactions, scoring system, security measures, and integration with external APIs.

1.4 Precedence and Criticality of Requirements

Requirements will be prioritized based on their criticality and importance to the core functionality of the bot. Critical features such as score prediction submission and scoring calculation will be prioritized over secondary features.

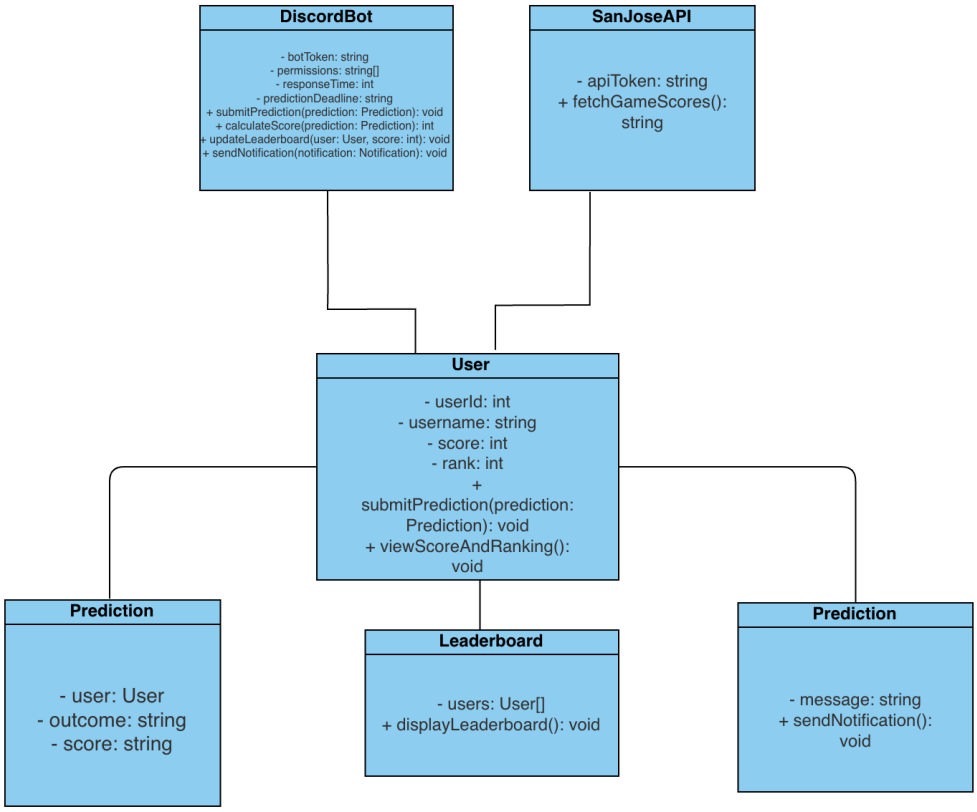
1.5 Evaluation Criteria

The success of the San Jose Earthquakes Football Prediction Discord Bot will be evaluated based on user engagement, accurate scoring calculations, seamless integration with the Discord platform, and ease of use for users of the server.

1.6 Acronyms and Abbreviations

- API: Application Programming Interface
- Discord: Discord Communication Platform

1.7 UML Diagram



2 Functional Requirements

2.1 Score Prediction Submission

Users should be able to submit their score predictions for upcoming San Jose Earthquakes football games through Discord. The submission process should be user-friendly and intuitive.

2.1.1 Predictions Timeframe

- Predictions are open to users for a limited amount of time
- Users cannot submit predictions once the game has started (or close to kickoff).

2.2 Scoring Calculation

The bot should calculate scores for users based on the accuracy of their predictions compared to the actual game scores and goalscorers.

2.2.1 Points Allocation

- Users earn points for correctly predicting the outcome (win, lose, or draw).
- Users earn additional points for guessing correctly the specific score (1-1 score for example in the match)
- Users earn additional points by predicting the first goalscorer (player on the Earthquakes team to score first in the match)

2.3 Leaderboard Display

The bot should display a leaderboard that ranks users based on their accumulated points.

2.3.1 Leaderboard Features

- Users can view the leaderboard at any time using a specific command.
- Leaderboard will show the top 5-10 users with the most points

2.4 User Engagement Features

To enhance user engagement, the bot should include additional features.

2.4.1 Game Notifications

- Bot will send notifications about upcoming games
- Bot will notify the timeframe of when users can predict
- Notifications include game start times and prediction submission deadlines.

2.4.2 Users Score Viewing

- A command allows users to check their current score and ranking.
- Will allow viewers to check their individual ranking and score

2.4.3 Squad Details

- Users can see the starting lineup for the game
- Users can also see which players are currently injured or healthy

2.4.4 Upcoming Matches

- Users can see the next matchday/opponent and the type of competition the match is being played in (League Game or Cup Game)

3 Performance Requirements

3.1 Response Time

The bot should respond promptly to user commands and notifications.

3.1.1 Response Time Constraint

- Bot should respond within 2 seconds of receiving a command

3.2 Scalability

The bot should handle a large number of users during peak prediction submission times.

3.2.1 Concurrent Predictions

- Bot should handle concurrent predictions for at least 200 users

3.3 Scalability

3.3.1 Runtime

- The bot should be running at all times

4 Environment Requirements

4.1 Development Environment Requirements

The development environment must meet specific criteria for successful bot development.

4.1.1 Discord Bot Token

- A valid Discord bot token is required for integration with the Discord platform.
- The Discord bot token should be stored in a safe place and not be available for the public

4.1.2 API Integration

- Integration with the Football API is required for fetching real-time game scores.

4.1.3 Database

- Need to store the predictions and information in a database to compare results to tally up and distribute points to the users.
- Will use Sequelize to specify the format/define the models of the data represented in javascript/typescript and will connect it to a relationship database such as PostgreSQL

4.2 Execution Environment Requirements

The bot must run in a stable and secure environment for public use.

4.2.1 Hosting Platform

- The bot will be hosted on a reliable server with continuous uptime.

4.2.2 Discord Server Permissions

- The bot requires specific permissions on the Discord server, including message sending, user information retrieval, and reaction capabilities.

4.2.3 Discord User Permissions

- Will outline clearly what information the bot will take in and store which is the username and predictions for the games throughout the season
- Want to highlight that no personal information will be taken
- Getting permission for the bot to be allowed in the server

5 Future Enhancements

5.1 A.I Component

The potential addition of an A.I component is being discussed and something that, assuming we get the main features of the bot working, could add further features to the project.

5.1.1 Predictive Analysis

- Will try and predict the upcoming score of the match using historical records such as head-to-head data

5.1.2 A.I Command Feature

- Users can type in a command to have the same predication as the bot, essentially agreeing with the bot's predication

5.1.3 A.I Participation

- Since the bot will be able to predict matchups, the bot will also participate in the leaderboard and compete against other users

idk