Bradley Bares & Joe Huang
Prof. Kathleen Durant
CS3200 - Database Design
12/8/21

CS 3200 Project Report

Specifications:

Spikeball is a game played between two teams of two players. The aim of the sport is to return the ball to the net utilizing 3 touches alternating between the two players. When a team unsuccessfully returns the ball to the net the other team gains a point. The first team to reach a predetermined score wins that game.

Spikeball tournaments are run with one to many divisions (skill levels). A division has many teams sign up for it. A team can only sign up for one division. A division has many matches amongst teams within a division. Each match consists of one to many games where the team that wins the majority of the games wins the match.

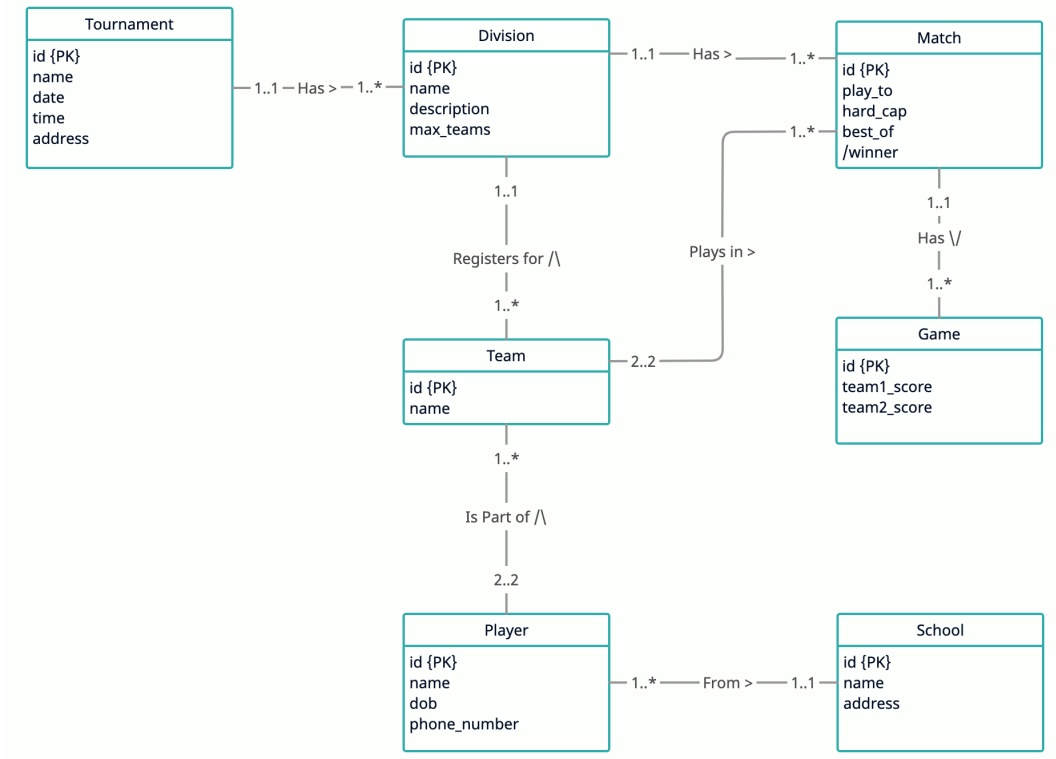
Tournament Manager is a native Python application leveraging Python's Tk GUI toolkit Tkinter and using MySQL interface PyMySQL to communicate with the designed SQL storage database. The app allows users to create/update/delete tournaments, tournament divisions, division teams, division matches, and game scores
within a match.

README:

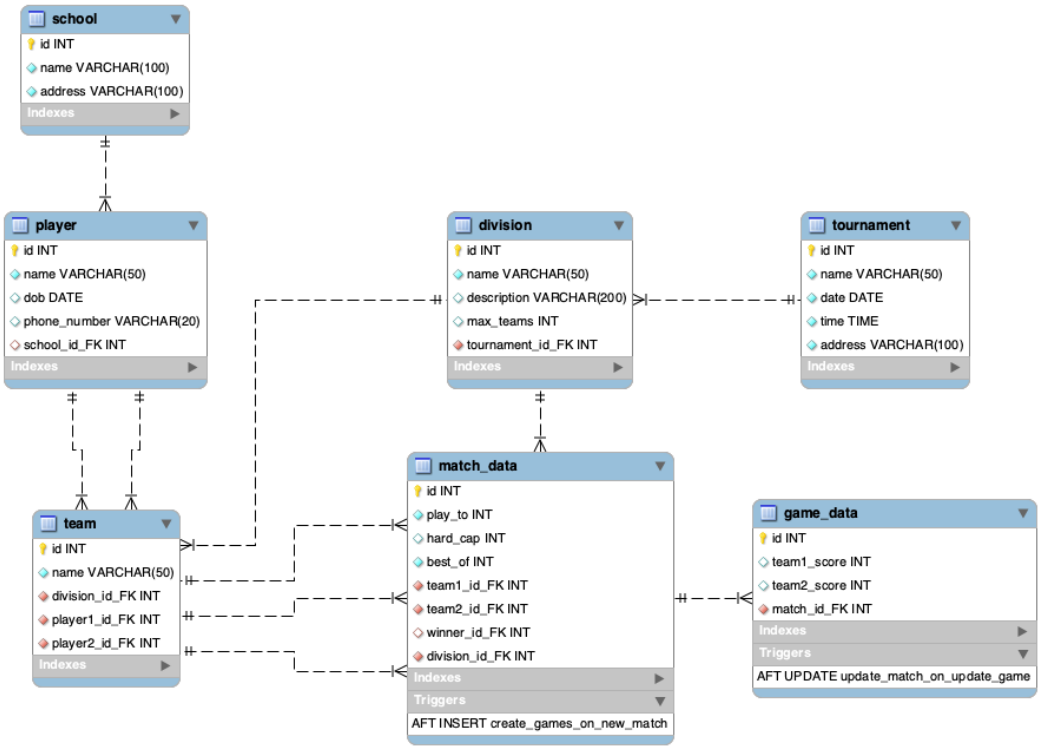
Most systems will have Python installed, to verify your version run `python3 --version` in your command line. If your version is lower than 3.6 or the command does not return a version number, visit the Python downloads page (https://www.python.org/downloads/) and download the latest version. Tkinter comes pre-installed with the Python installer binaries, but if you don't have Tkinter installed you can run `pip3 install tk` in your command line. PyMySQL does not come pre-installed and can be installed by running `pip3 install PyMySQL` in your command line.

Utilize the included database dump `dump.sql` to setup the MySQL database using your preferred database management tool. Adjust `host` and `db_name` in `main.py` based on the results of the previous section. Finally navigate to the application directory and run `python3 main.py` in your command line.

Conceptual Design:

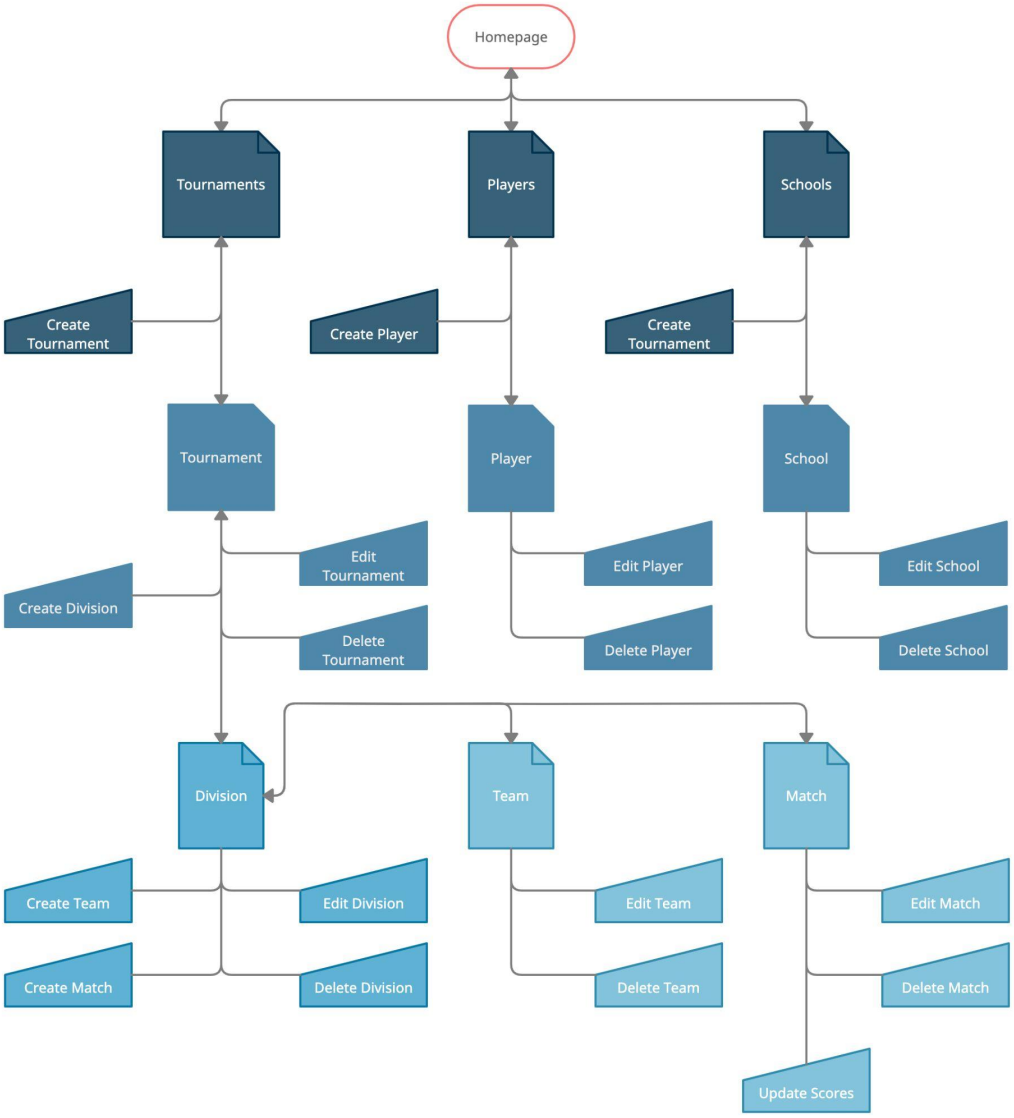


Logical Design:



User Flow:

First, the user is brought to a login page where they must type in their username and password to login. This then brings them to a page that gives them options on whether they want to look at tournaments, players, or schools. They can click a button to go to the tournaments page. This gives them the option to create a new tournament by typing in its information, or selecting a specific tournament to look at. When a tournament is selected to be looked at, it's information from the database is read and displayed. The user is also given the option to edit the information of the current tournament being looked at. This page also lists all of the subunits of the current tournaments such as the divisions, matches, or teams. The user is able to click on the subunit and repeat the same operations mentioned previously for tournaments, but for that unit of data. All of these operations are generalized and repeated for the various units of Spikeball data such as the ones mentioned earlier like people, or schools, as well.



Lessons Learned:

1. The technical expertise gained in this project includes how to design a MySQL database so that it's usable in a practical software application, how to design a functional GUI in Python using its graphical libraries, and how to interact with MySQL databases using Python.
2. An insight on time management during this project was realizing how time consuming it is to learn a graphical library from scratch in order to create an application. Beyond this, it's notable how much time was needed to edit oversights in our project such as database design that wasn't compatible with our frontend. This resulted in time being spent editing the conceptual and logical design of the database.
3. The only alternatives designs we considered within this project were primarily rooted in MySQL. This was because we were often unsure how we wished to represent various aspects in Spikeball due to how it's a rather unique and detailed sports system. This resulted in some complex data representations that we ultimately scrapped in favor of more streamlined designs.

Future Work:

1. We plan to use this database to gather information on future Spikeball tournaments so that it's easier to keep track of various statistics.
2. Potential areas for added functionality three fold. The UI/UX can be improved by changing the styling of widgets, adding stat tracking presentation, and offering more ways to input a type of data. The database can be improved by comparing functionality with a NoSQL database. Finally the codebase can be improved by adding more abstractions for code segments and revisiting abstraction decisions made at the beginning of the project, now with more information available.