## Entity-Relationship Diagram

**User entity attributes:** Email, I.D (underlined/key), Games Played, Game#, Wins

**Game entity attributes:** UID1, UID2, GameNum (underlined/key), Board, GameOn, Winner, Turn, Moves

**Relationship:** User —— Play —— Game

### User

| UID | Email | GameNum | TotalWins | GamesPlayed |
|-----|-------|---------|-----------|-------------|

### Game

| GameNum | UID1 | UID2 | Board | GameOn | Winner | Turn | Moves |
|---------|------|------|-------|--------|--------|------|-------|

---

Users will provide an email and password at login for their account. Once logged in a surrogate key I.D will be made. This will keep track of a users total wins and games played number.

Email: Varchar(??)
I.D: Int (Primary Key)
Game#: Int, defaults null
Wins: Int, default null
Games played: int, default null

Once a game is started, two users I.D will be put together along with a surrogate game number will be made. A string representing the board will be made and the game begins. The turn shows the UID of who is currently up and the moves keeps track of how many moves have been made. Presumably once the moves made hits five, an algorithm needs to be made to see if we have a winner. If there is a winner, a report is sent to both emails and the winner I.D gets a win added and the row from the game table is deleted.

GameNum: Int, not null, Primary Key, Foreign Key On update cascade: On delete cascade
UID: Int, foreign key, not null, on update cascade, on delete set null
Board: Varchar(9), default (Whatever our begining string is)
Game on: bool, default true
Winner: Int (UID) default null
Turn:Int (UID)
Move: Int, default 1