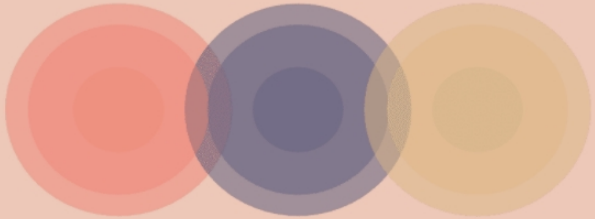


In a row

Table of contents

In a row

Introducing the core concept of our application

II

In a row

Part 1 | A brief overview of the application

01 The idea behind the game

02 How the game works

03 What we thought to include

Rules & Features of the game

- Same rules as the classic retro game - 4 in a row, also called Connect 4!
- Connect 4 pieces of the same colour in one line, regardless of the direction.
- Added functionalities like 3 in a row and 5 in a row.
- Opt for 1v1 or tourney games.

How we came up with idea

II

In a row

Part 2 How we came up with the idea

How we organised ourselves

1

Met once or twice a week

Decision made

2

The communication and group dynamics

What we used to collaborate

Zoom, Trello, Google Docs, Adobe Xd, Whatsapp Group etc

3

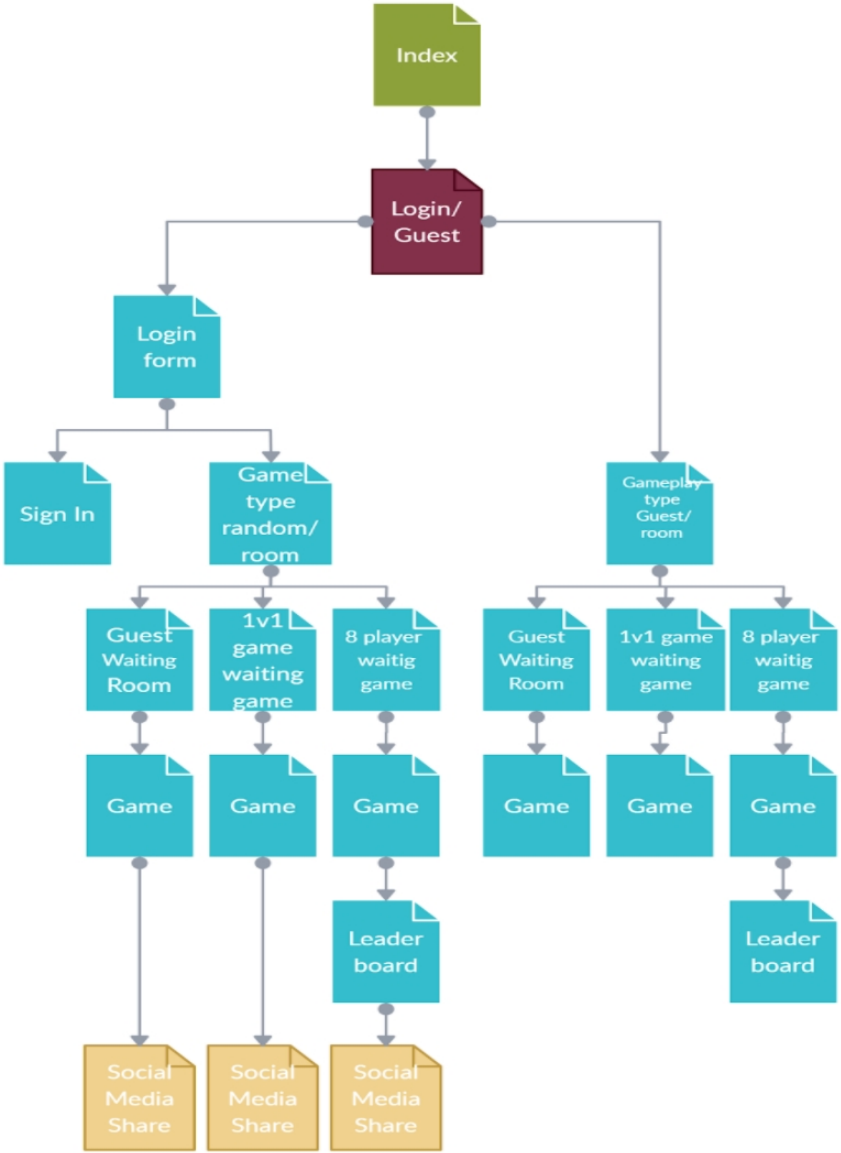
Different apps

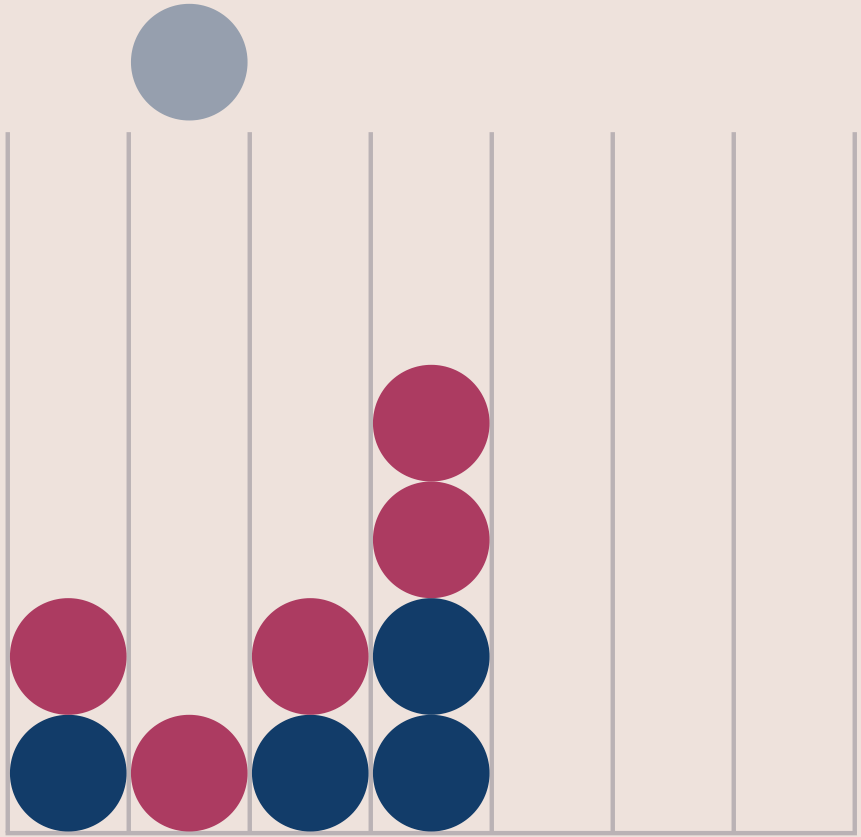
In a row

Design of the application

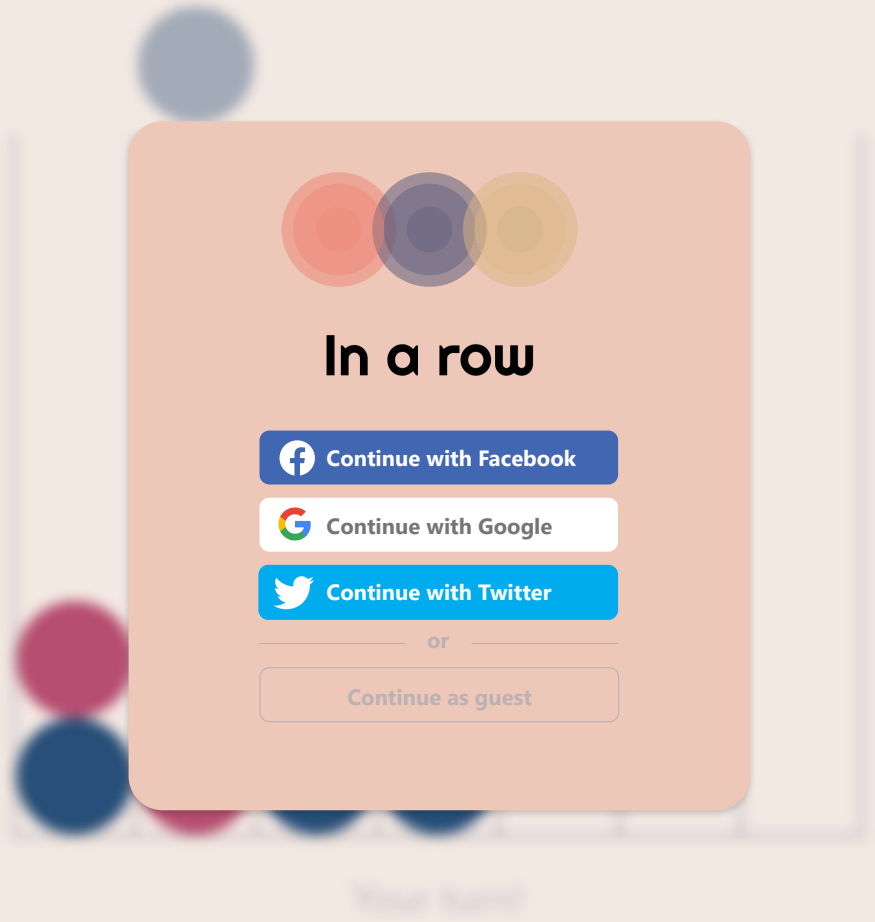
II

1 Flow Chart explanation





Your turn!

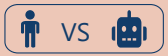
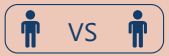

In a row
Continue with Facebook
Continue with Google
Continue with Twitter
or
Continue as guest

Setup a game!

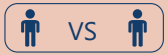
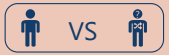
What in a row:

3 in a row | 4 in a row | 5 in a row

Local game:

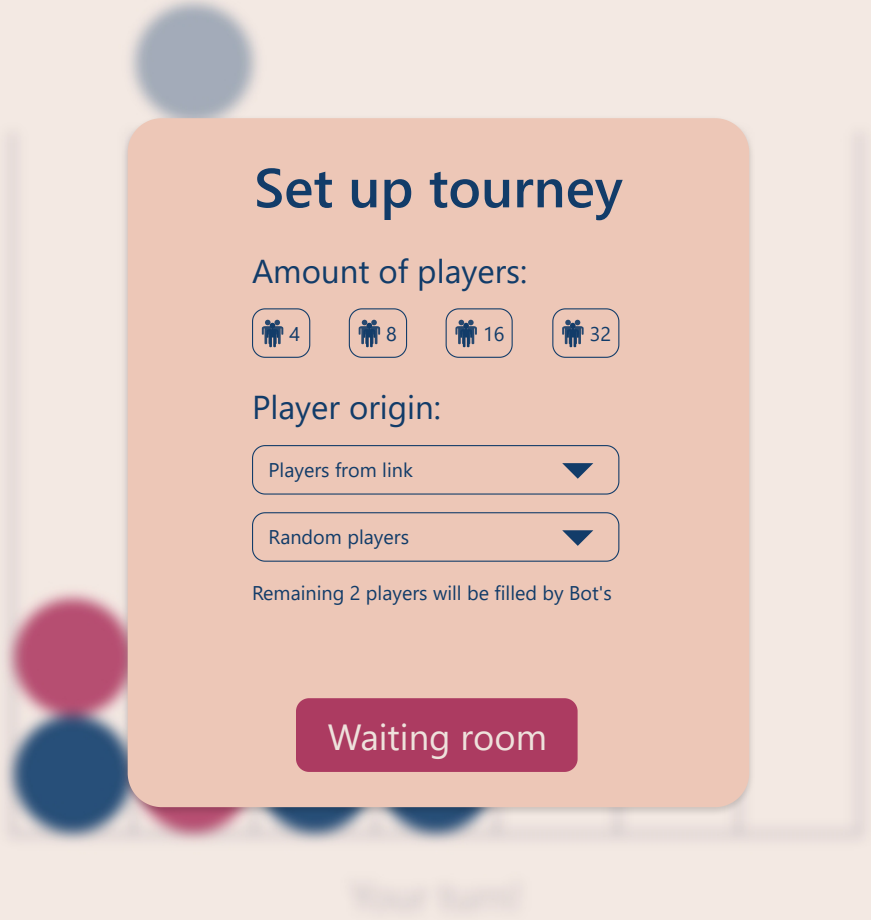
 

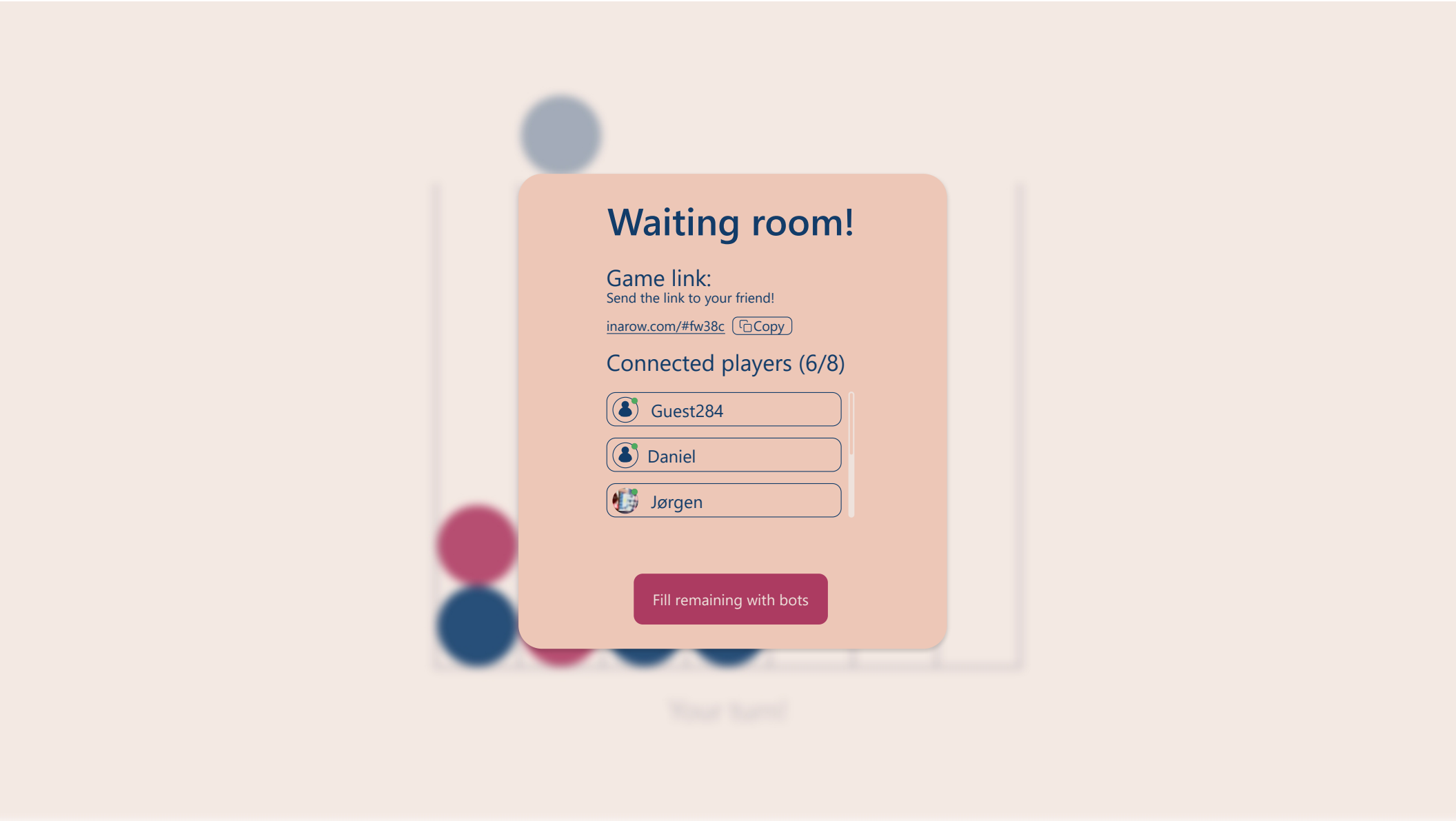
Online game:

Let's play!


Set up tourney
Amount of players:
4
8
16
32
Player origin:
Players from link
Random players
Remaining 2 players will be filled by Bot's
Waiting room


Waiting room!
Game link:
Send the link to your friend!
inarow.com/#fw38c
Copy
Connected players (6/8)
Guest284
Daniel
Jørgen
Fill remaining with bots

Get ready!

Round 1:

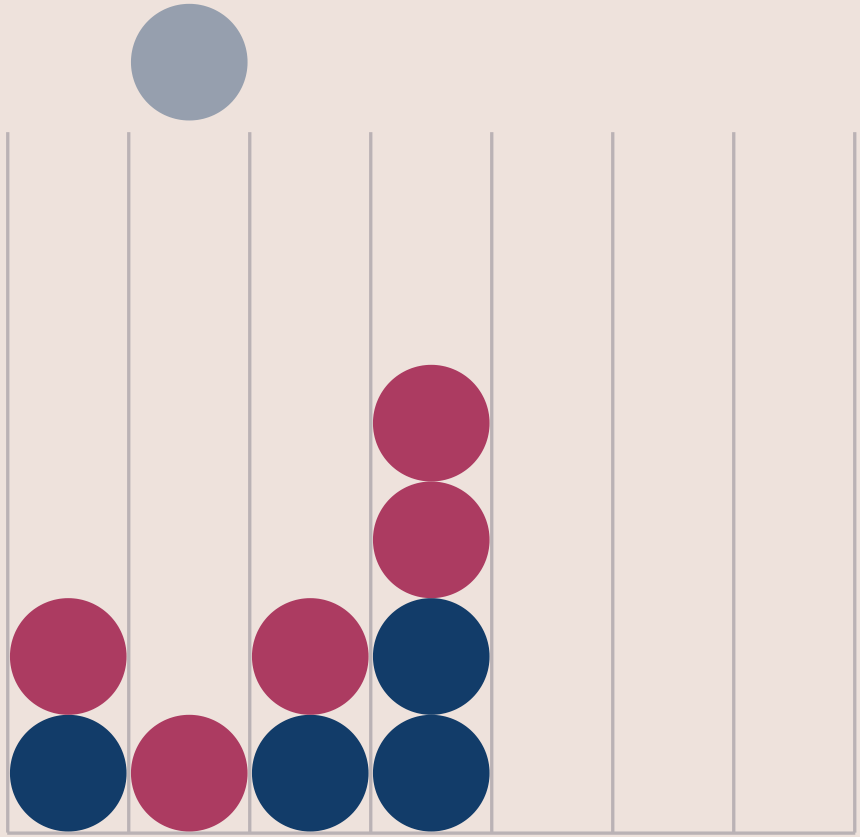
Guest284

VS

Jørgen

Guest321
James
Jørgen
Janet
Laura
Guest384
Guest321
Milly
Guest321
3rd
Jørgen
Axel
Milly
5th
Guest384
Janet
Axel
7th
Guest384

Leaderboard

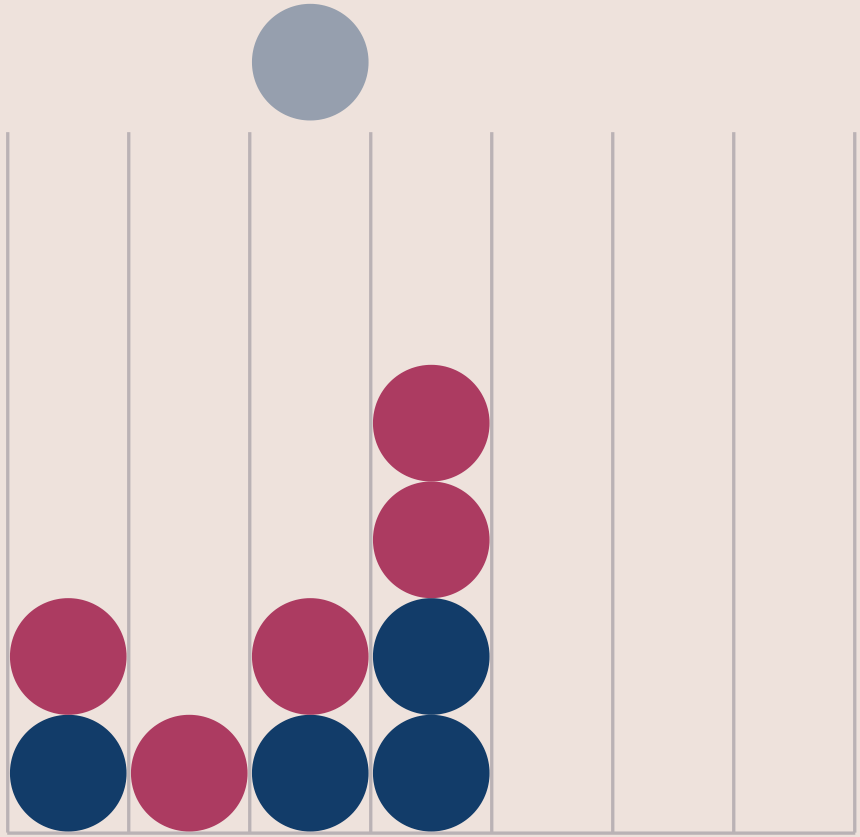


Your turn!

Help

Guest321
James
Jørgen
Janet
Laura
Guest384
Guest321
Milly
Guest321
3rd
Jørgen
Axel
Milly
5th
Guest384
Janet
Axel
7th
Guest384

Leaderboard

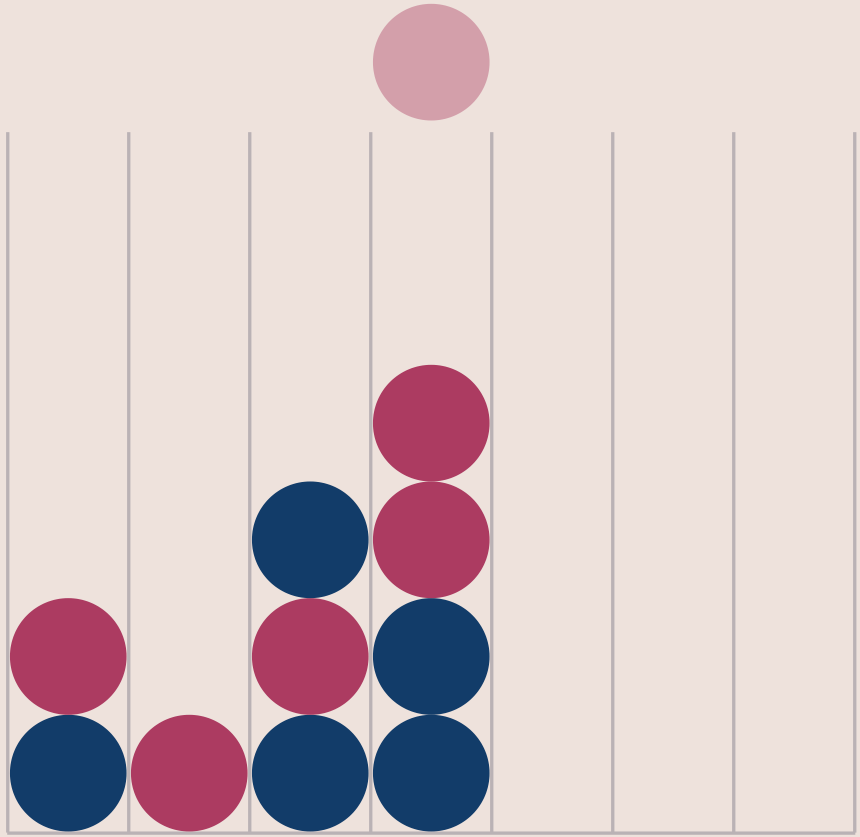


Your turn!

Help

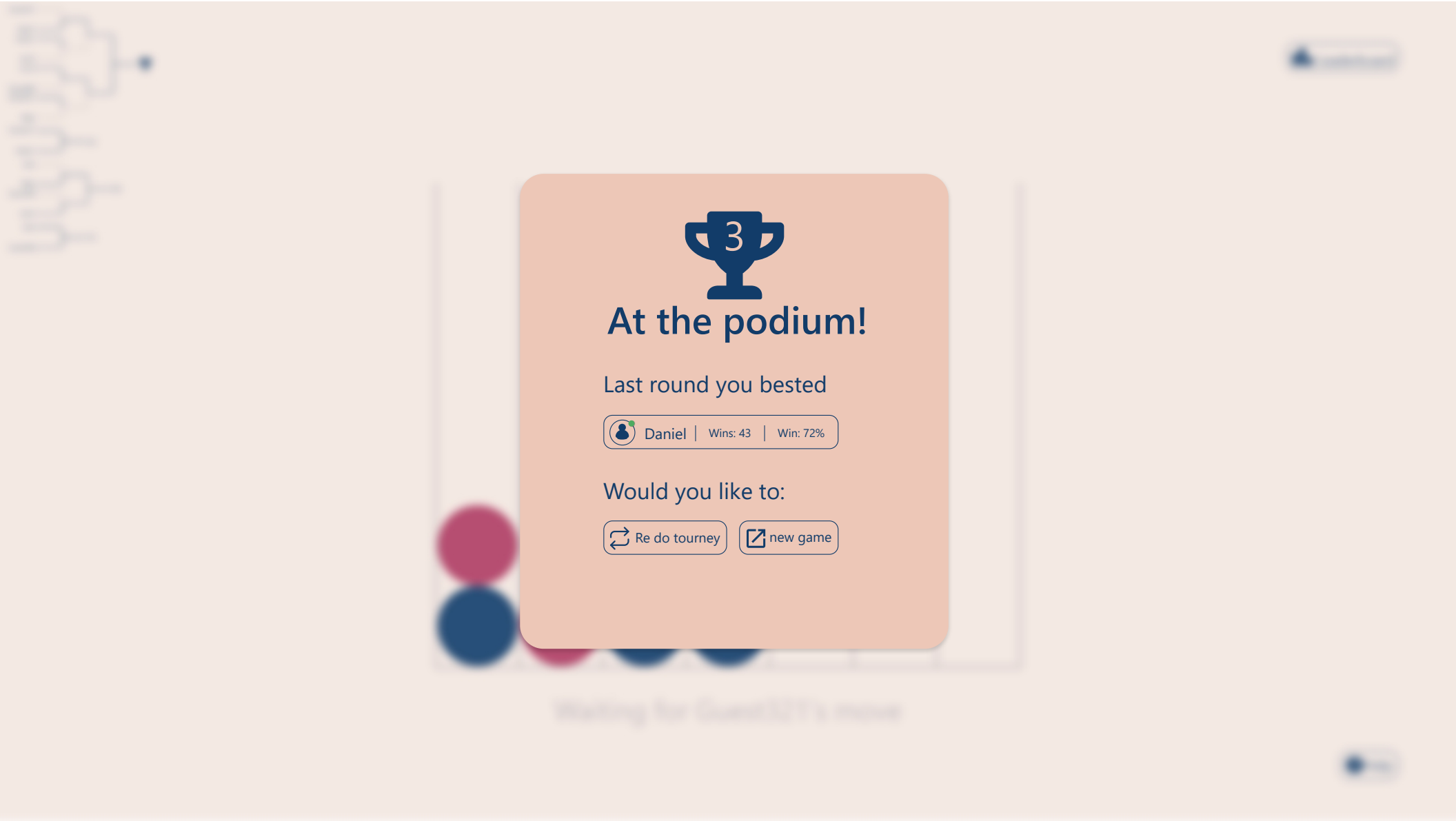
Guest321
James
Jørgen
Janet
Laura
Guest384
Guest321
Milly
Guest321
3rd
Jørgen
Axel
Milly
5th
Guest384
Janet
Axel
7th
Guest384

Leaderboard



Waiting for Daniel's move

Help


3
At the podium!
Last round you bested
Daniel | Wins: 43 | Win: 72%
Would you like to:
Re do tourney
new game

Part4	Design of the app

2

The design in general

1. **Strive to achieve good user interaction**

2. **Flat UI design trend**

#EDC7B7

#EEE2DC

#BAB2B5

#123C69

#AC3B61

In a row

Part4 Design of the app

2

The design in general

1. Strive to achieve

2. Flat UI design tre

In a row

Continue with Facebook

Continue with Google

Continue with Twitter

or

Continue as guest

In a row

Game Complexity and the logics behind it

II

In a row

Part 5 Complexity of the game

Database

01

02

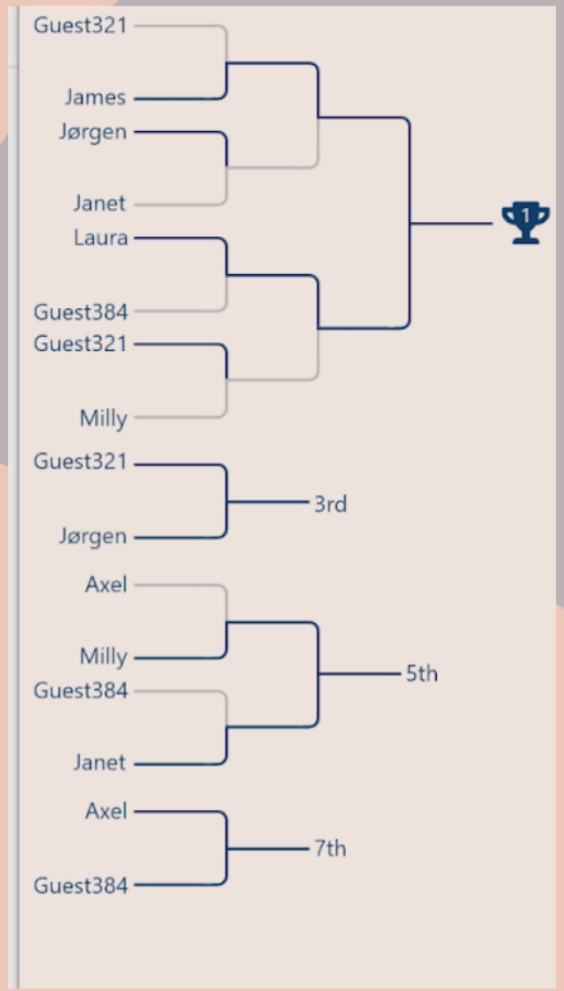
Structure of the tournament

03

User experiences

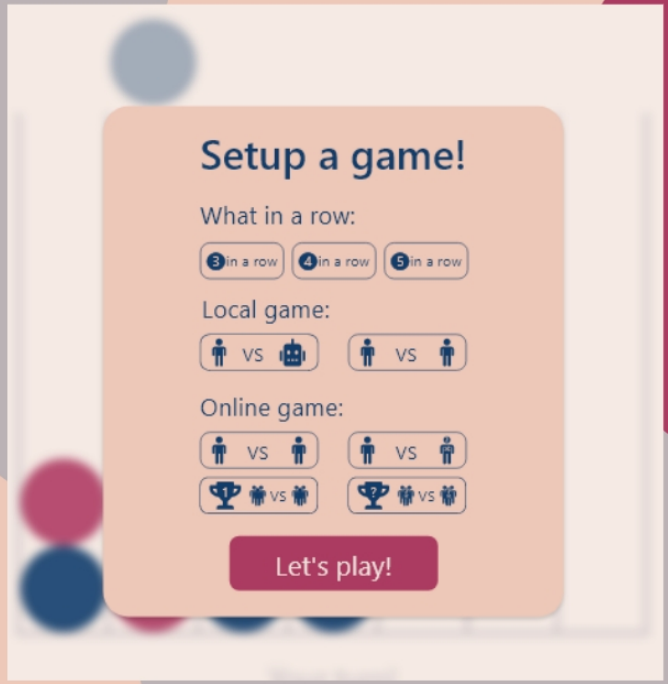
Leaderboard, Win rate, Facebook/Sign/Login In Option, play with bot, play with a random person

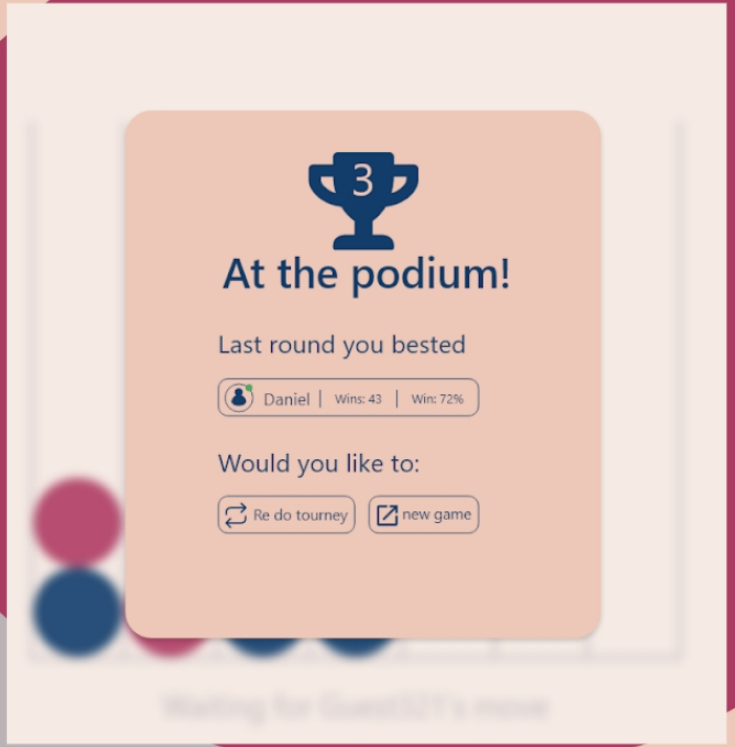
In a row



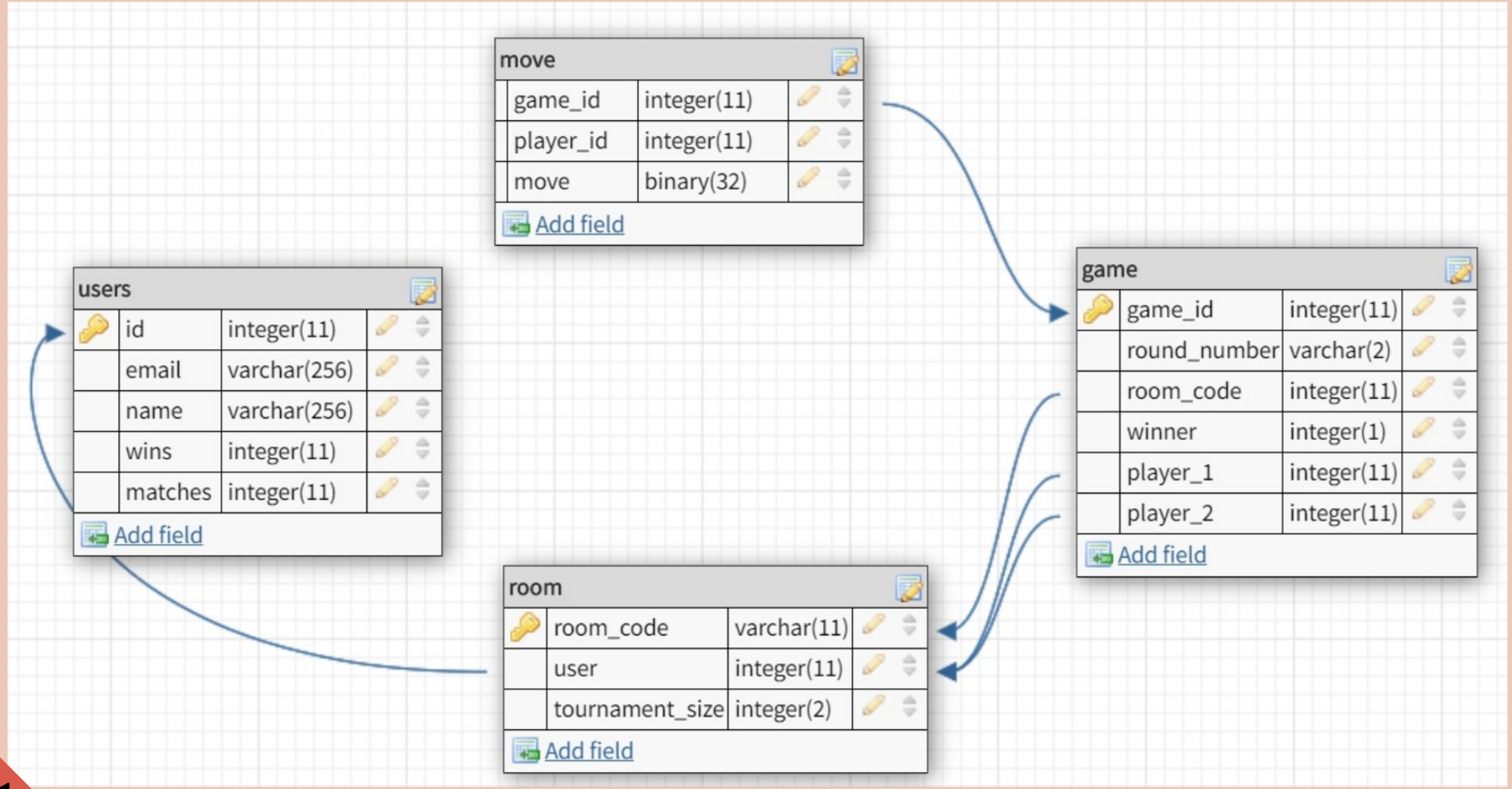
Tournament Structure example for 8 players

Game setup options





Leaderboard example


move
game_id integer(11)
player_id integer(11)
move binary(32)
Add field
users
id integer(11)
email varchar(256)
name varchar(256)
wins integer(11)
matches integer(11)
Add field
game
game_id integer(11)
round_number varchar(2)
room_code integer(11)
winner integer(1)
player_1 integer(11)
player_2 integer(11)
Add field
room
room_code varchar(11)
user integer(11)
tournament_size integer(2)
Add field

Implementation of the game

II

Why Node JS for Back-end Development

- Good choice for applications which have to process a lot of information and require low latency.
- Very lightweight due to basic event-driven structure.
- Therefore, easy to maintain and modify.

In a row

Why React for Front-end Development

- Simplicity in picking-up syntax;
- Flux controls flow of data through, single entry point, making it easy to debug;
- Descriptive.

Why MySQL for our database

- It's a free-to-use, open-source database.
- Globally renowned for being the most secure and reliable DBMS.
- Offers high performance due to its distinct storage-engine framework.
- Simplistic and straightforward, ideal for applications which don't require complex data structures and offers seamless integration.