SECURITY

Users should not be able to access each other’s accounts or edit each other’s property (teams, leagues).

In cases where two parties are needed, A request system is set in place. An example is the leagues <-> teams connection. League managers can change the link in the table, but only the team managers can insert rows(requests) into the leagues <-> teams table.

Another example is the teams to user’s link. In order for a user to join a team, he/she must obtain the code from the team lead in order to join. This key is hashed in the database.

Everything is hashed. Users passwords to sign into their accounts, keys needed to join teams, keys needed to edit organizers information, and so on.

Time was not spent covering injections. Users can easily inject js into text fields on their profiles. This was not addressed do to the time crunch on finishing the endpoints.

Poor overall structure gives admins breaking power. Also, makes no sense from anybody looking in. Sometimes I don’t think it makes sense.

Everything is only hashed once. And I built almost everything. Therefore, it is not safe.