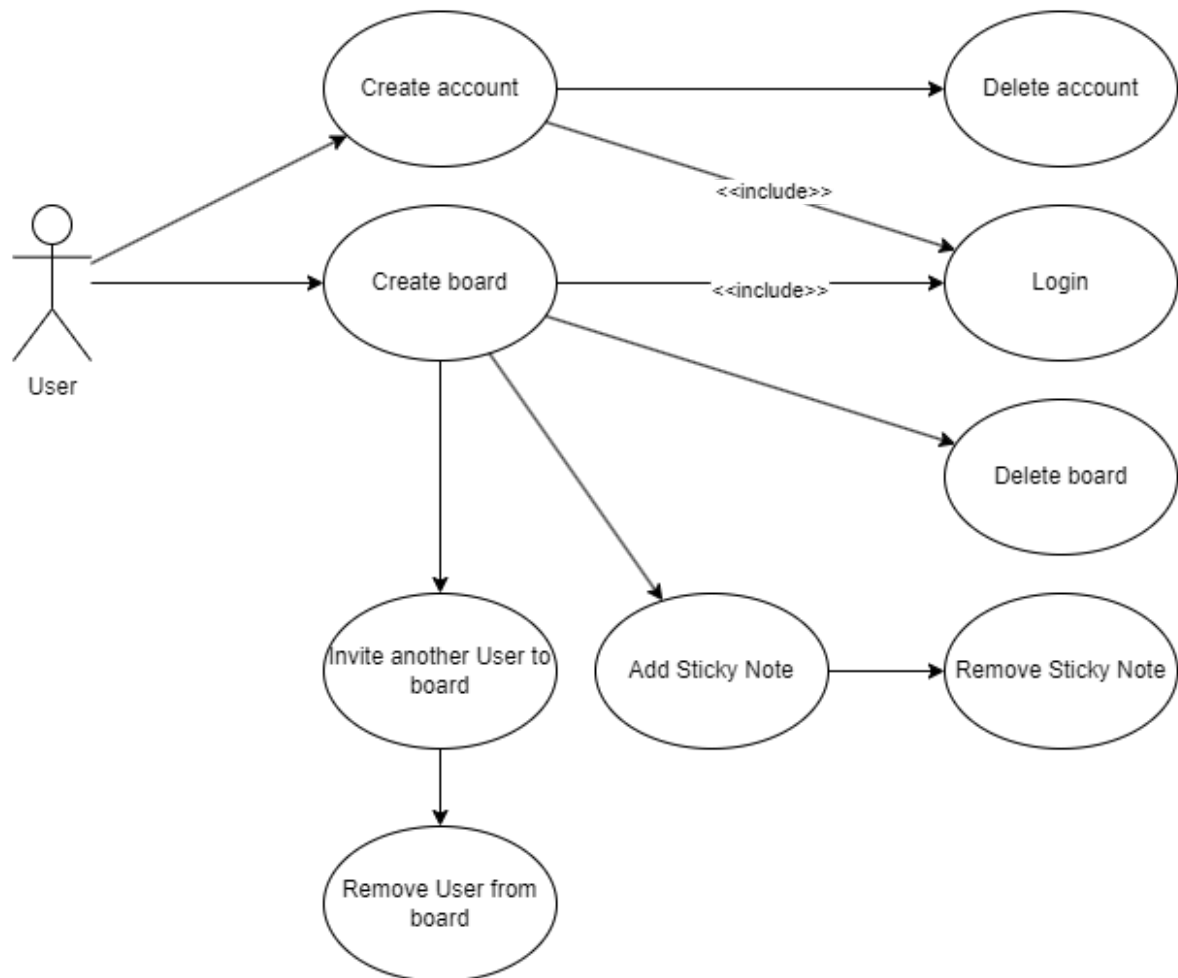


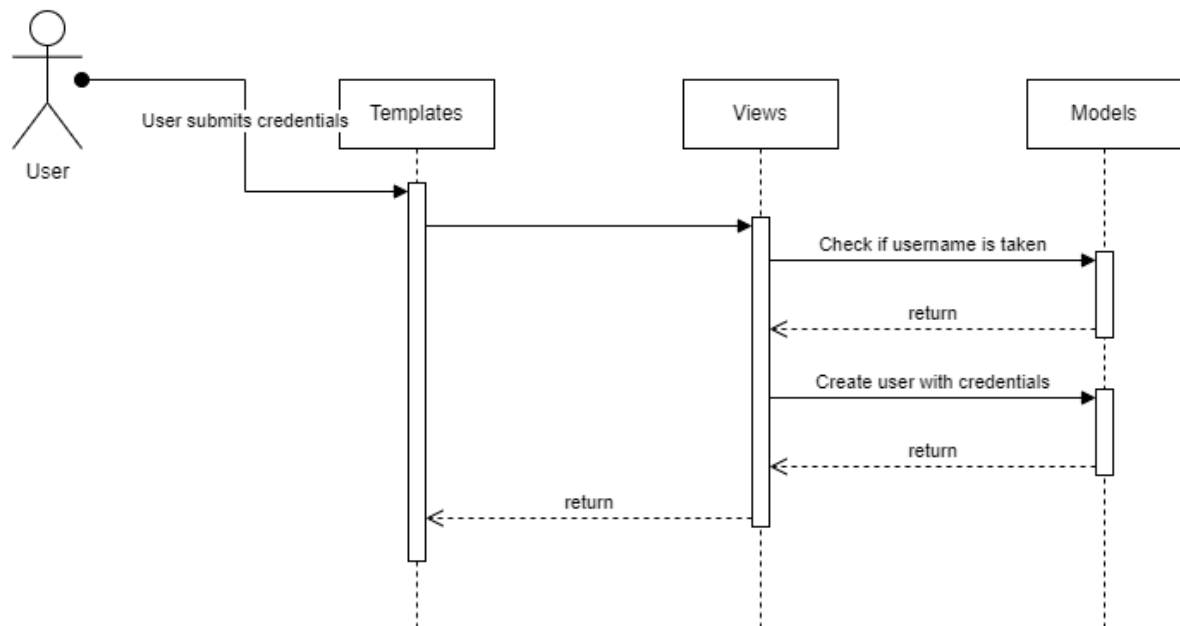
Sticky Notes Application Design

Use case diagram

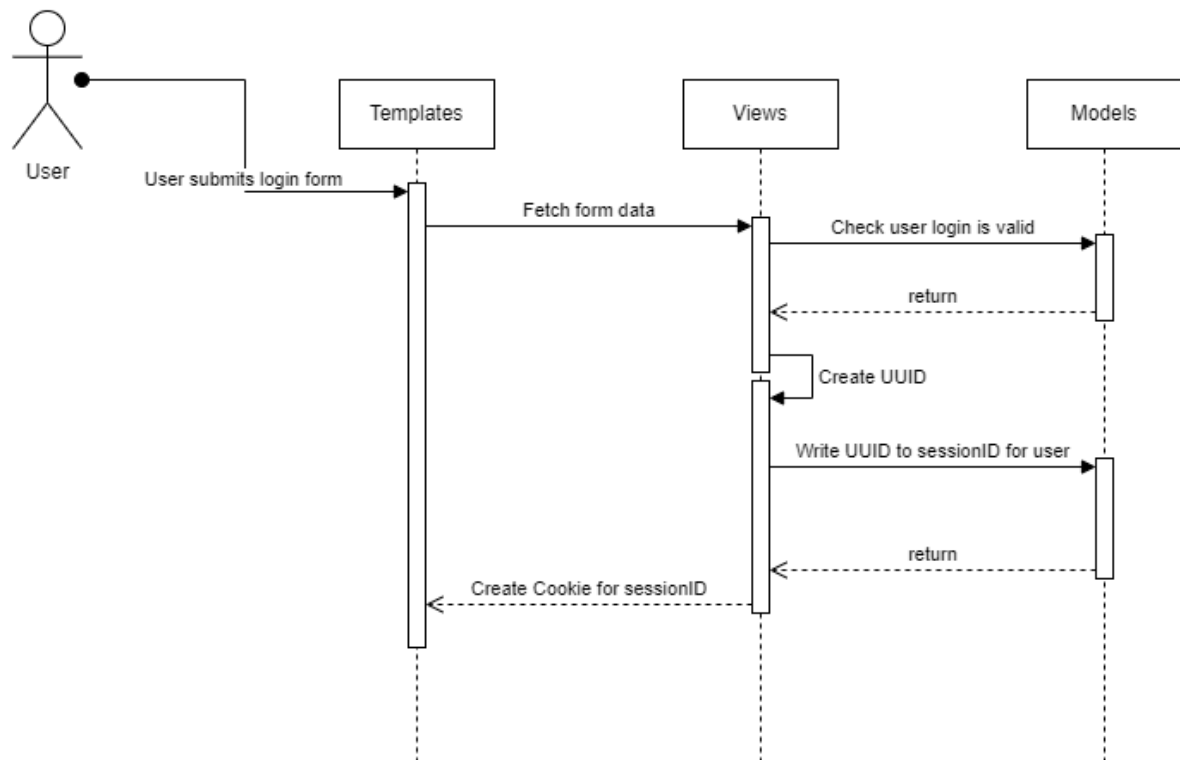


Sequence diagram

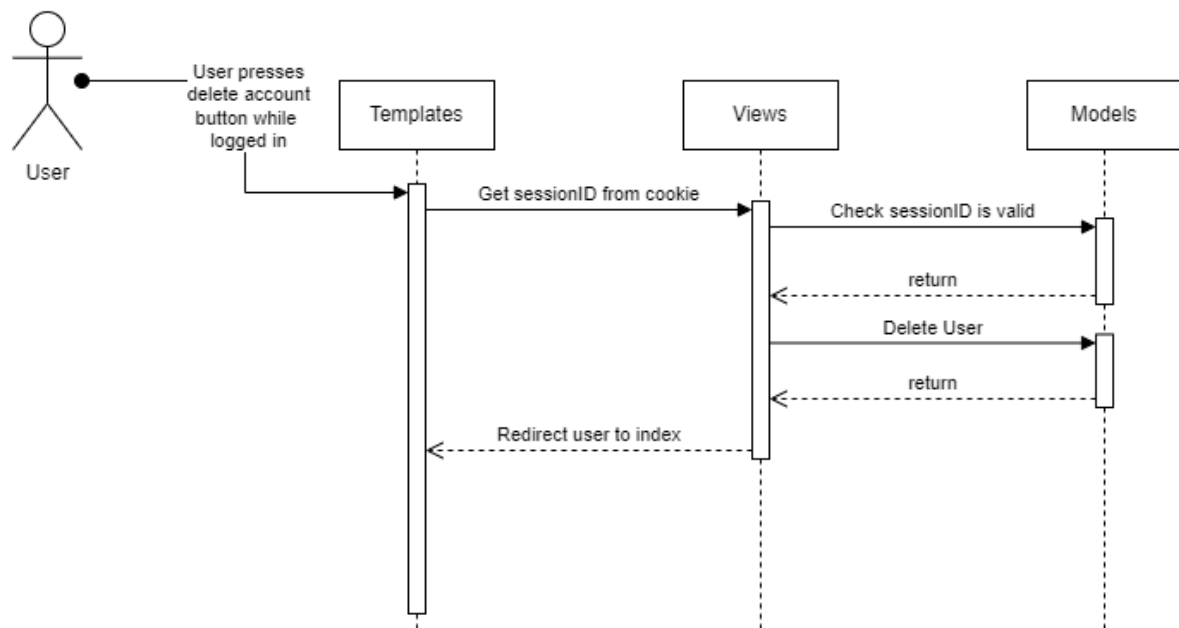
Creating account



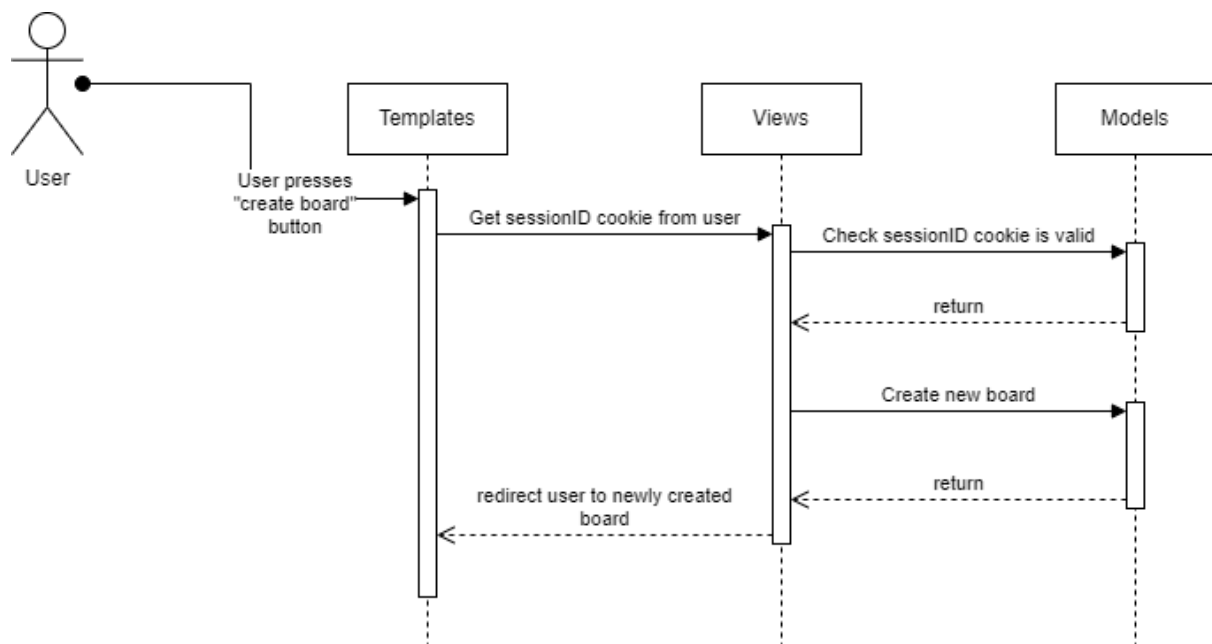
Logging in



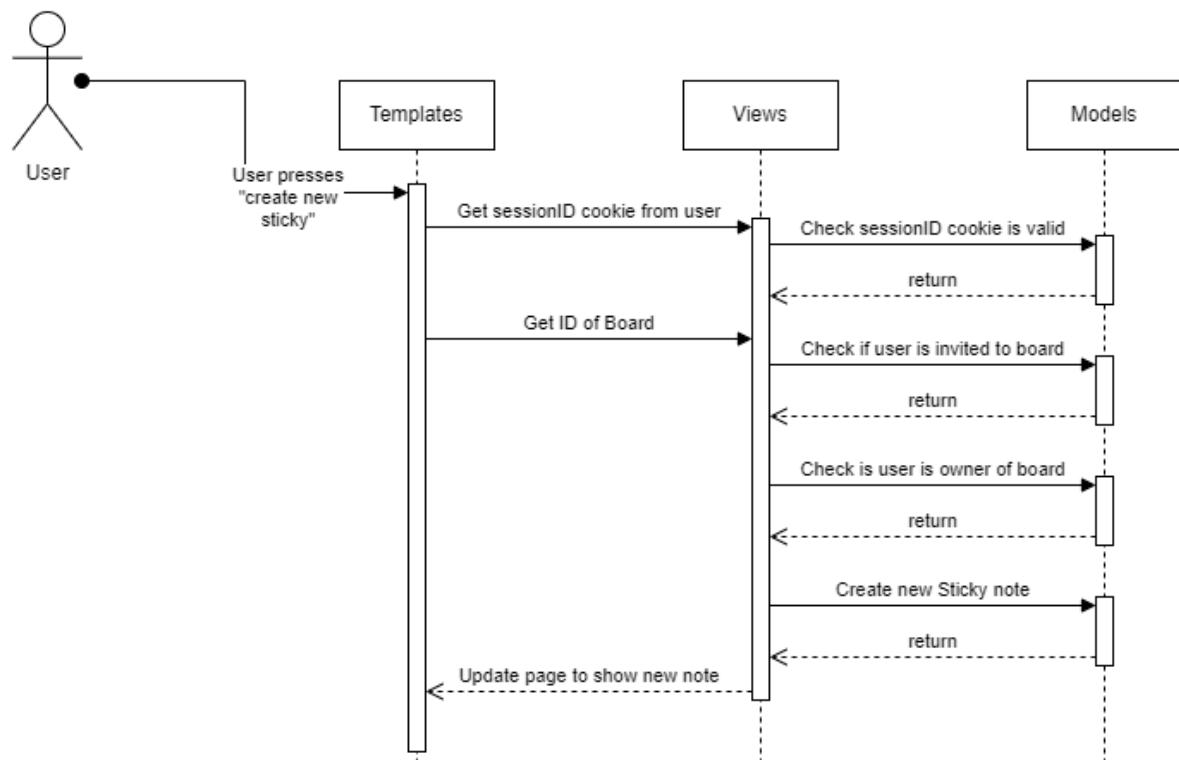
Deleting account



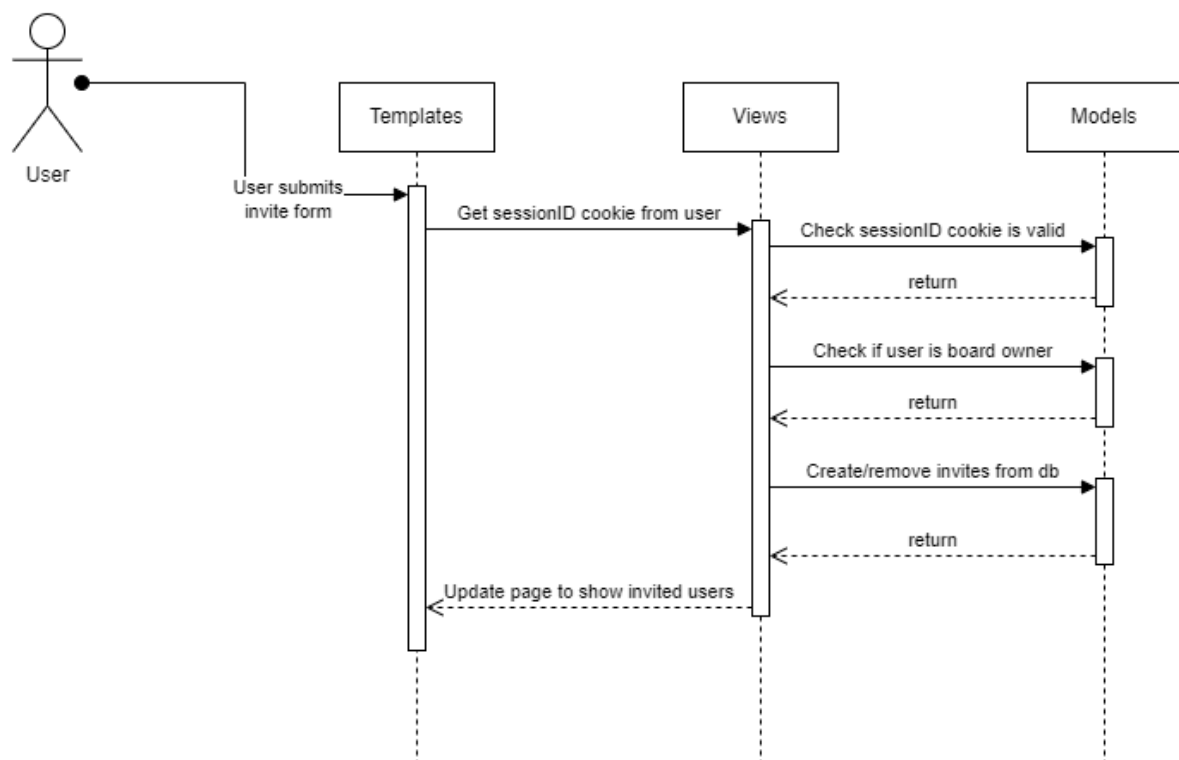
Creating/deleting a board



Adding/removing sticky note

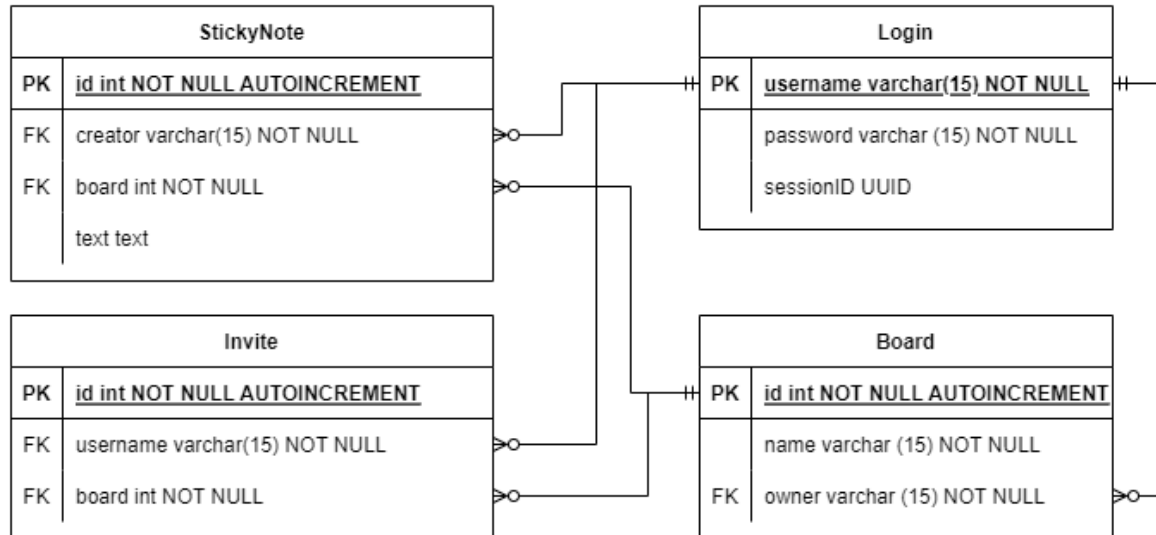


Inviting/removing user from board



Class diagram

Models



User session preservation

When a user enters the correct credentials, the server creates some sort of ID or hash, which is then sent to the user in the form of a cookie. When the user needs to do something that requires a login, instead of sending the credentials every time, the server checks the user's cookie to see if the ID in it matches up with the one it created. In my case I generate a UUID4 and send it to the user and store it in the database ([seen here](#)).

Configuring server based DB

To do so all that is needed is to edit the DATABASES constant in the settings.py file, so that django connects to the right endpoint and with the right credentials, you just need to check the database type is supported by Django. Additional tinkering may be needed on the database itself to allow django access rights. If the settings are correct, when migrations are run, Django will modify the database according to your models.

Github Repo Link

<https://github.com/NSAJones/Sticky-Notes/tree/main>