

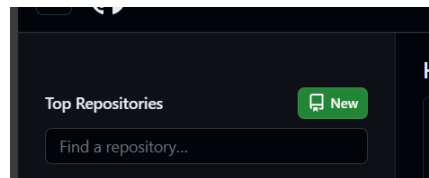
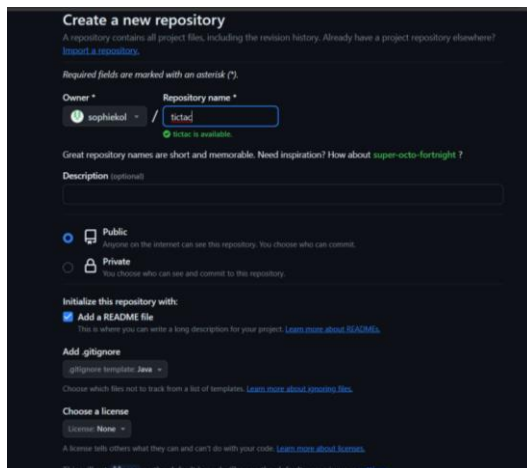
DOCUMENTATION TICTACTOE PROJECT

Link: <https://github.com/sophiekol/tictactoe>

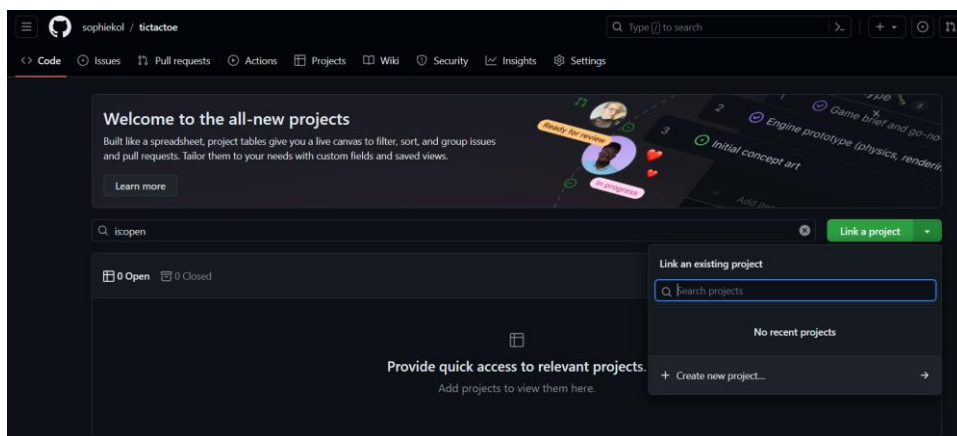
1) Create new repository on Github

Neues Repository auf Github erstellen unter Dashboard > New repository

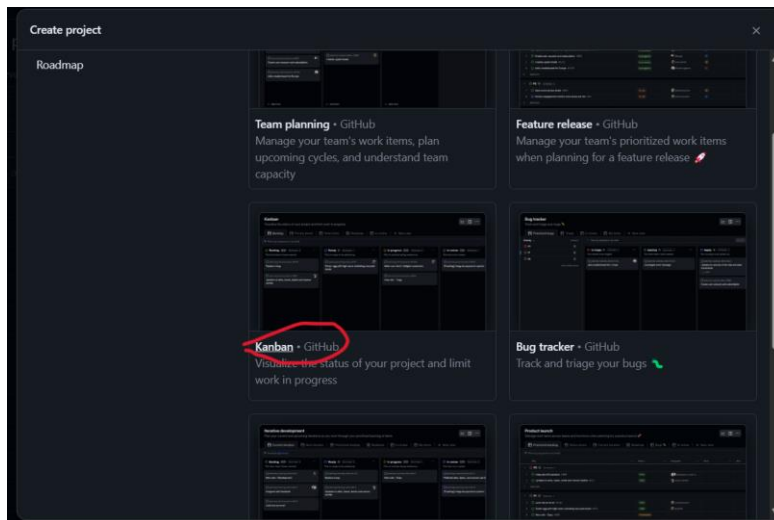
Repository benennen und auf public lassen, README hinzufügen und .gitignore



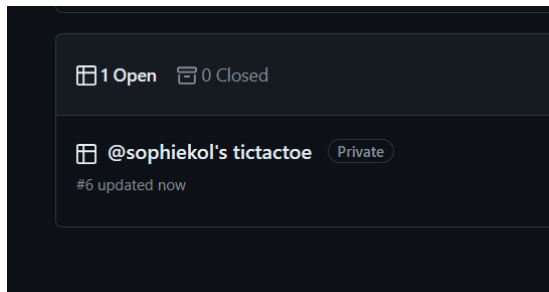
2) Neues Projekt erstellen und mit repository linken



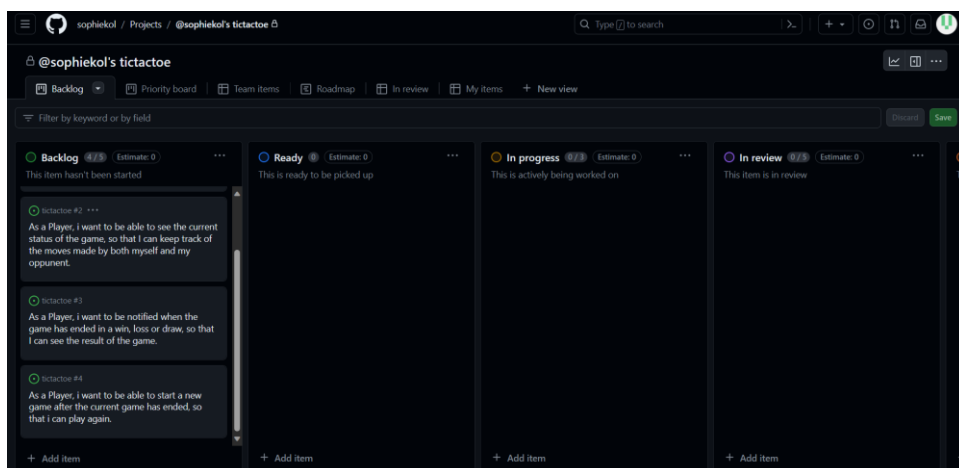
In demselben Project Template Kanban auswählen, damit man die User Stories Tracken kann:



Project ist jz mit unserem Repository gelinkt:



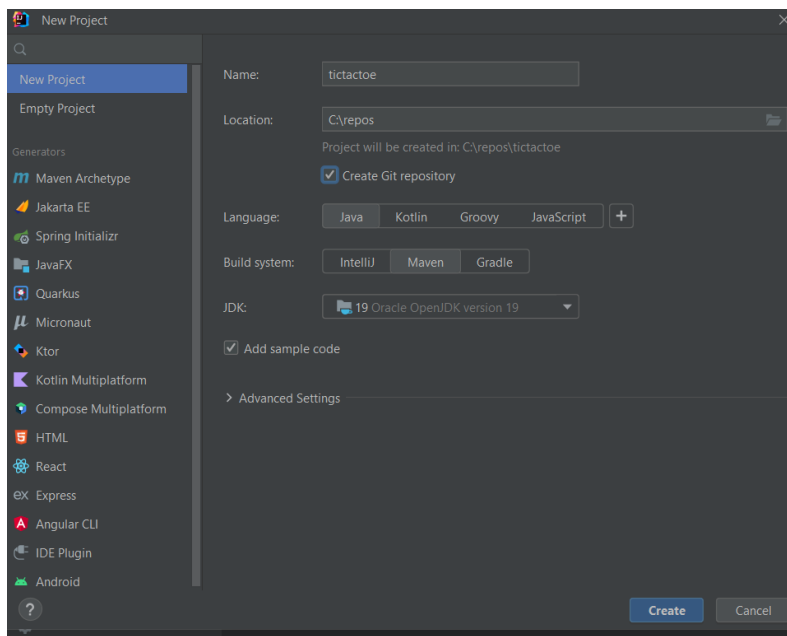
3) UserStories zu Kanban Board hinzufügen



Hier können wir jz den Process des Projektes verfolgen, UserStories markieren an denen wir gerade arbeiten, welche wir noch machen müssen und was wir schon erledigt haben.

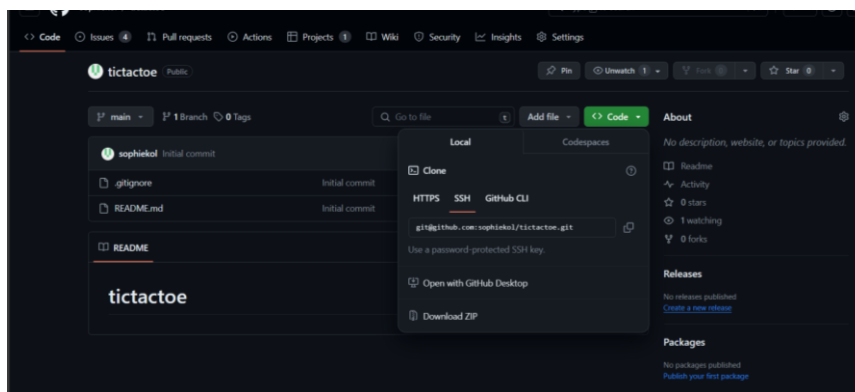
4) Erstellen von einem neuen Project in IntelliJ

Maven + local repository

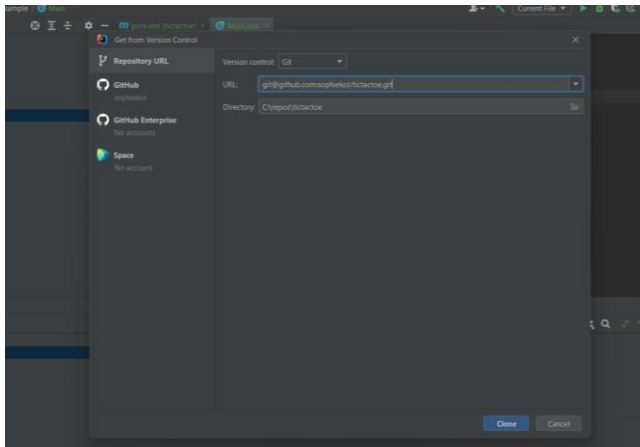


5) Clone Remote Repo from Github

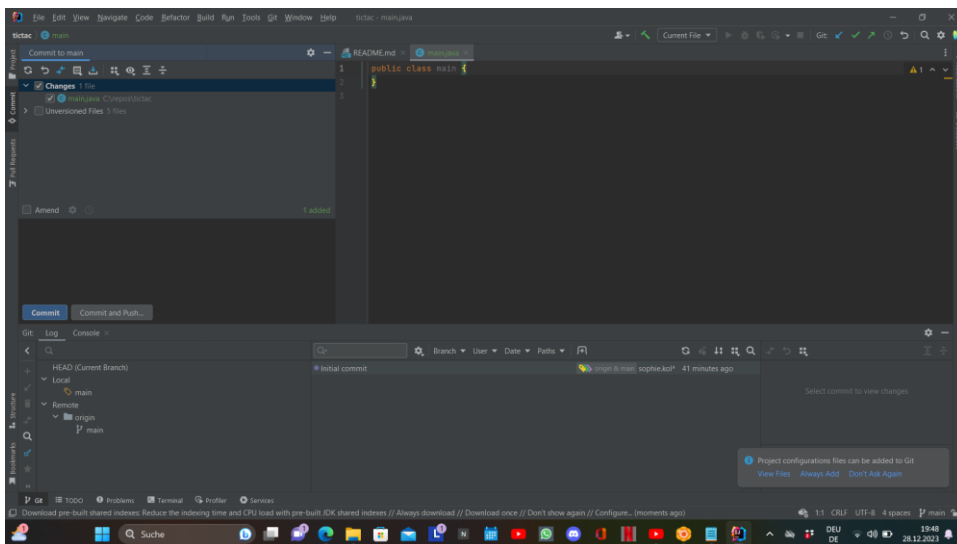
Code > SSH > Copy link to Project



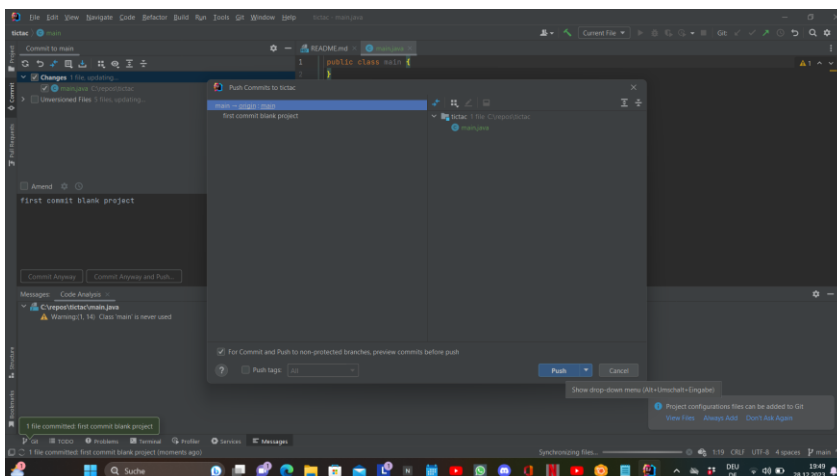
6) Einsetzen von link in IntelliJ



Netzt sieht man unten die Branches auf dem Local und auf dem Remote Repository.



7) Ersten Commit vom Blank project auf den Main Branch

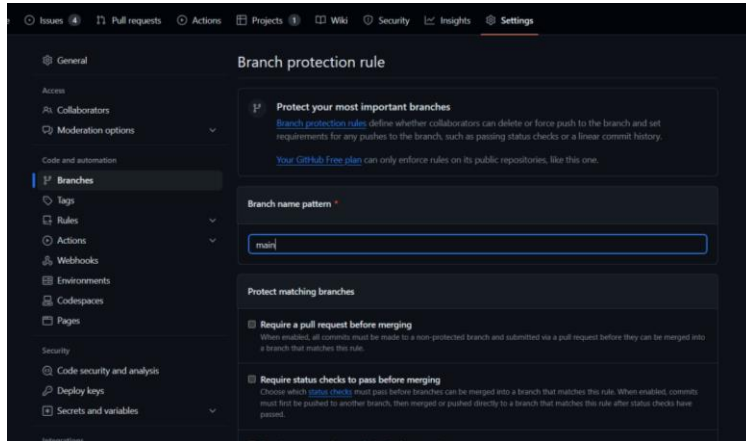


Pushen vom ersten Commit auf die remote Repository

8) Main Branch Protecten

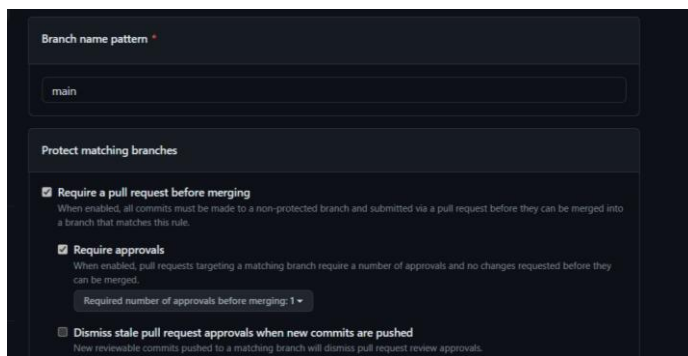
So einstellen das ein Pull Request entsteht bevor man etwas in die Main Branch mergen kann.

Im repo > Settings > Branches > name der branch eingeben für die man 'Rule' erstellen will



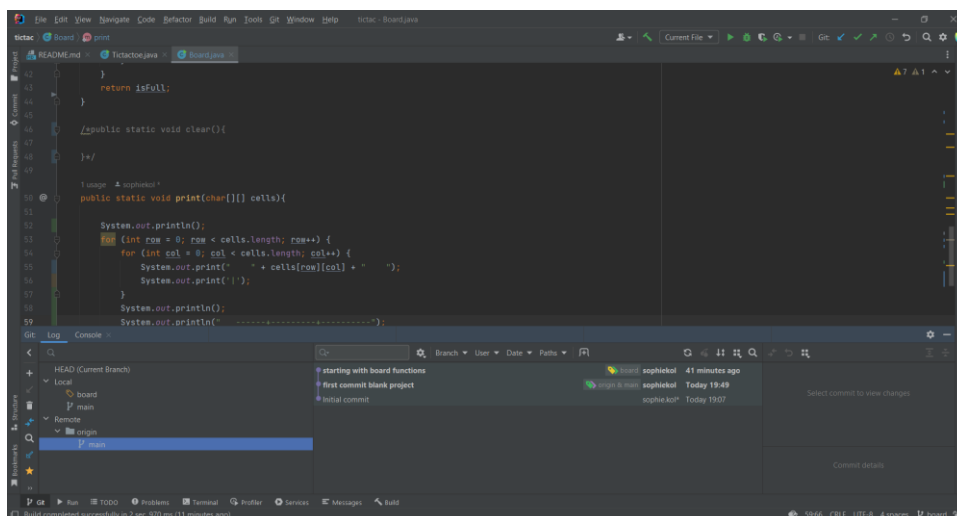
Auswählen: Require Pull request before Merging

Require Approvals:



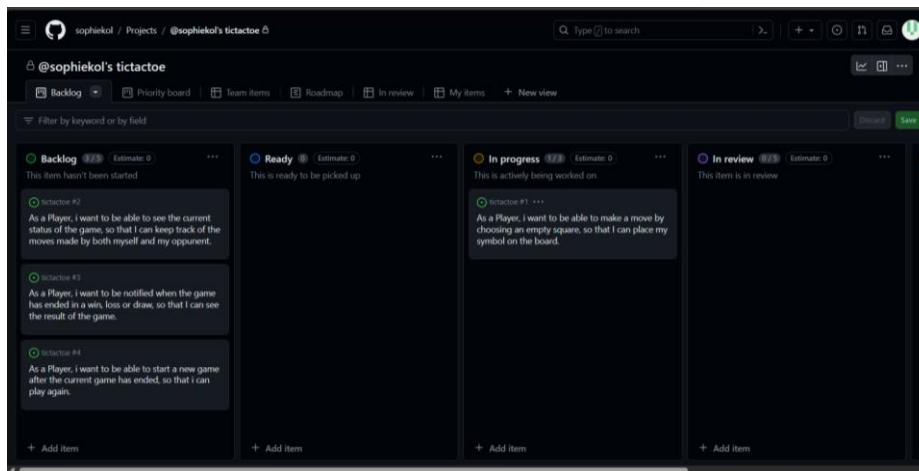
9) Anfangen am Project zu arbeiten

Commits auf die local repository laden

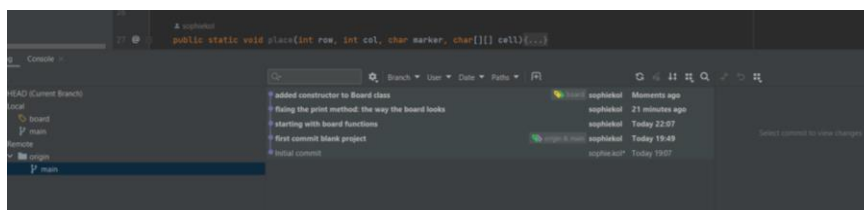


10) Kanban Boardupdaten

Board updaten, an welchem Feature gerade gearbeitet wird

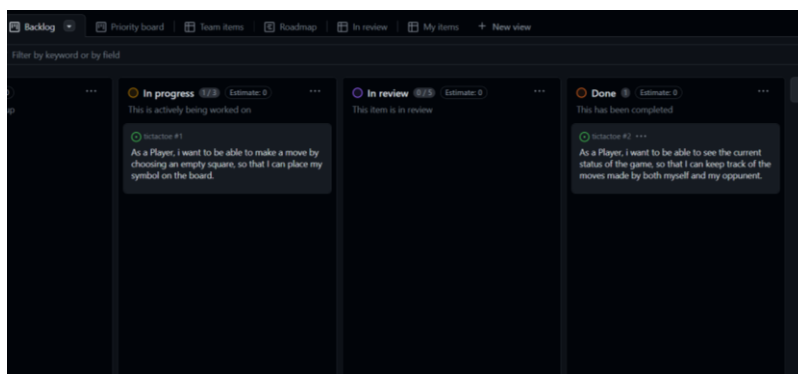


11) Local Commits weiterhin machen



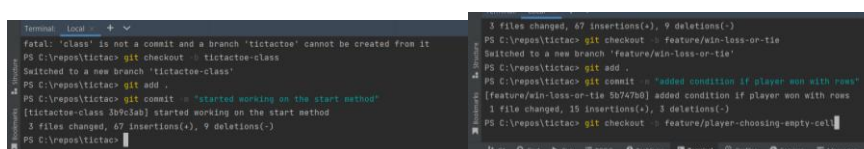
12) Kanban Board weiterhin Updaten

Bei jeden neu angefangenen Feature zum 'in Process' verschieben, Status of game wird in Done hin verschoben, da das als erstes fertig wurde

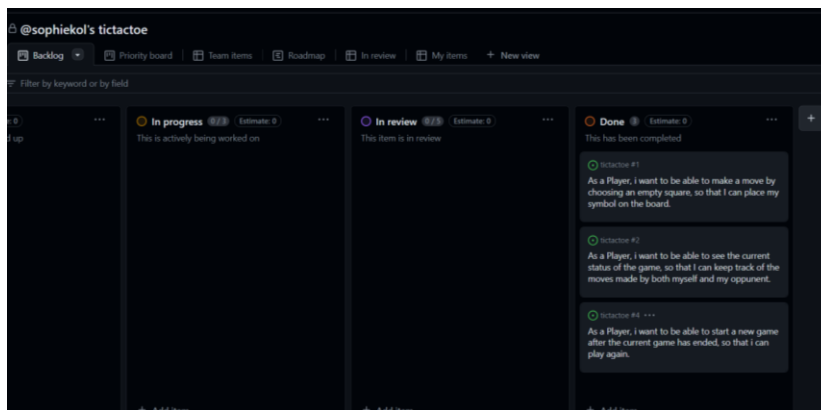


13) Neue Branches mit Git erstellen

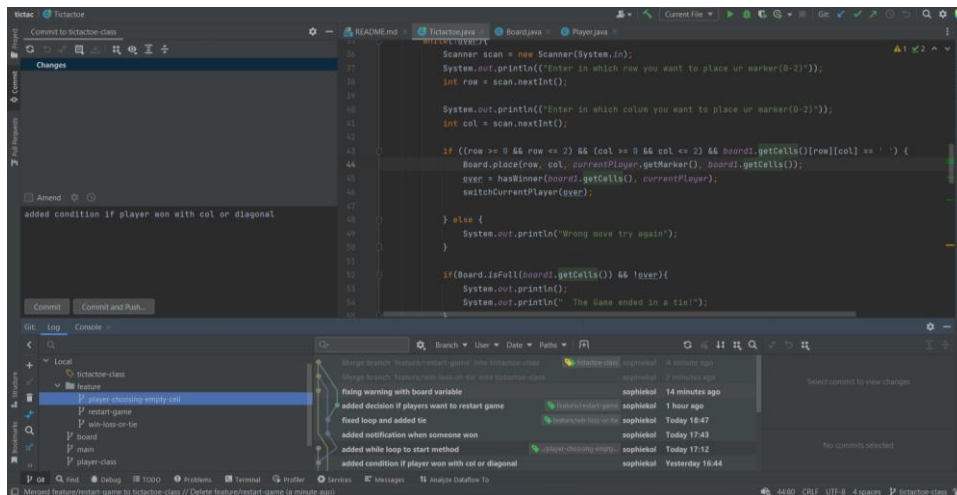
Für jedes feature einen eigenen Branch erstellen



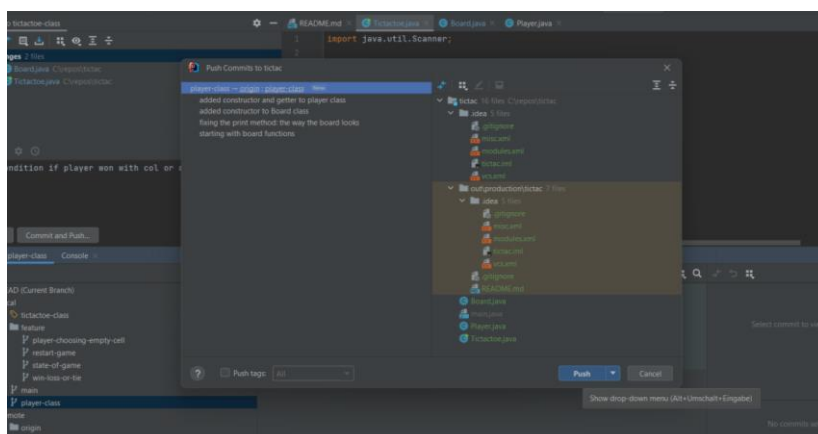
14) Prozess auf Kanban Bord verfolgen



15) Auf local repository mergen von branches



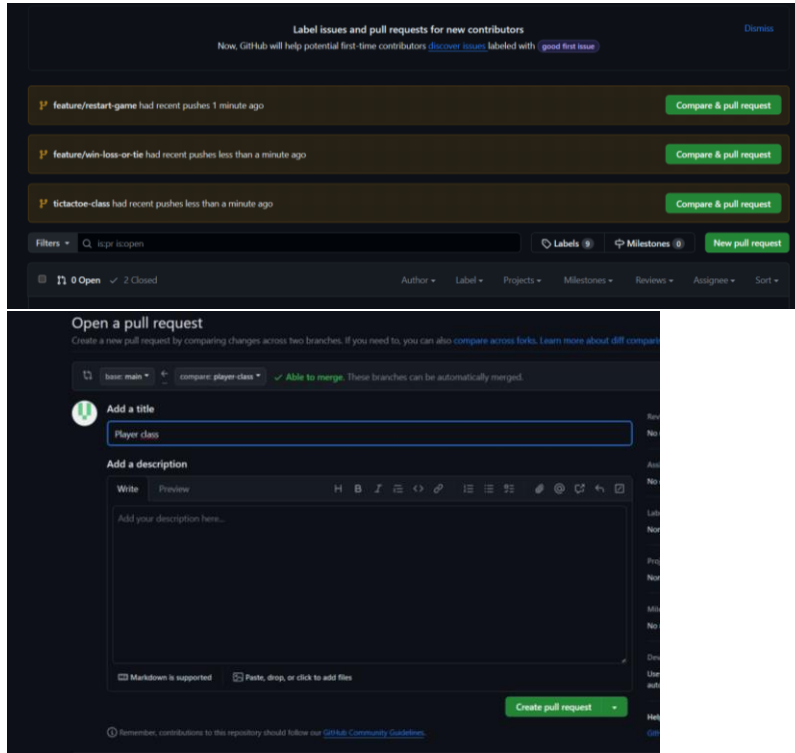
16) Erste Pushes auf Remote Repo vornehmen



17) Bei jedem Push Pull Request

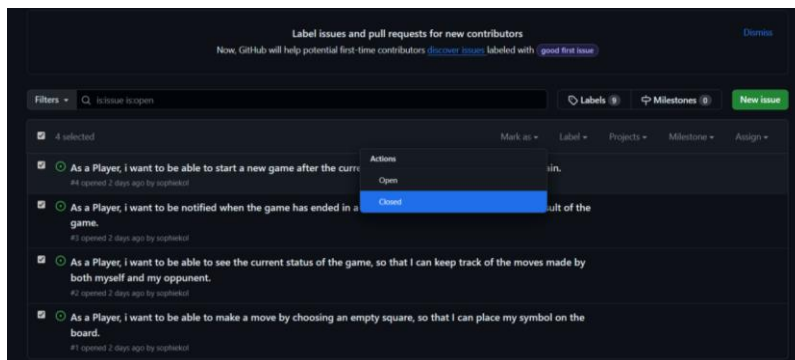
Bei jedem push wird ein pull request ausgelöst,

Nachdem die Features alle fertig sind werden die auch gepusht, man sieht die offenen pull requests:



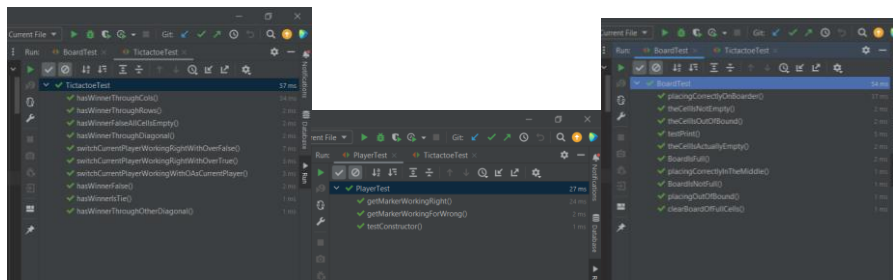
18) Updaten der Issues

Repo > Issues > close



19) Tests

Das Einzige, was noch gefehlt hat sind die Tests



Eigener branch wurde natürlich für die Testes erstellt