Feedback

Architecture

I think that you have achieved many of the grading criterias already. There are no visible bugs that I can find and you are using one file per view as well as using a good mix of views and presenters. From what I got from the code the chessBoard would count as a third party component which is a requirement for higher grade. To ensure getting a high grade just make sure you are not mixing any views with different roles and try to make the code UI-framework independent if aiming for an A-plus. All in all I think you have done a great job on the architecture. Something to think about if you want to implement could be an easier reversal of action (return button or similar).

Design

I think the design is looking promising as well. Even though we are only mid project I think you have a good foundation for the feel of the navigation through the pages. As a user I felt in control when navigating. The login/signup page was intuitive and browsing through the site felt very smooth. I also liked that there already was a consistent theme throughout the site. Something minor to think about could be the placement of the logout button, right now I think that it felt slightly hidden as it could only be seen in the settings page. Perhaps you could consider moving it to become more visible. Another thing you could be working more on is the homepage before logging in. Right now I understand what the site is about from the descriptive text but I think you could make it even more apparent using pictures etc.

Web-api

Not much to say here other than that your current structure seems promising. From what I can tell the authorization with firebase and state of the user seems to be working properly. Good job!

Github code:

- **Is the architecture following the principles you learned through the course and/or the guidlines of the framework the group is using?**

  Yes, I think it looks good.

- **Is the code DRY (Don't Repeat Yourself)? If there is code duplication (similar segments of code repeated in several places), they should have been extracted in a loop, method or module that can be reused.**

  From looking through the code on github I could not find any apparent segments where the code was repeated in an ineffective matter.

- **Naming convention: Are the files, methods and variables named in meaningful way (reflecting their purpose)? Is the same pattern used consistently throughout the code?**

  I think that there is a consistent pattern in the code structure and naming of variables/methods etc.

- **Code formatting: is the code properly indented and formatted for good readability?**

  Yes.