

Tekniska instruktioner

Jonas Teglund

August 2024

1 Teknisk bakgrund

ÅM-sidan är en webbapplikation skriven i React med Next.js ramverket. Den kommunicerar med en Firestore databas som innehåller all information om lag, grupper och matcher. Alla filer finns att hitta på Github: <https://github.com/jteglund/AM>

1.1 Koppla hemsidan till databasen

För att koppla Webbapplikationen till databasen behöver man ändra i `/tournament-manager/src/app/firebase-config.js`.

1. Hämta en api-config från firestore.
2. Byt ut `firebaseConfig` variabeln till den nya api-config.
3. Sätt `apiKey`-fältet till `process.env.NEXT_PUBLIC_API_KEY`
4. Sätt en environment-variable på maskinen som hostar applikationen: `NEXT_PUBLIC_API_KEY = api` nyckeln

1.2 Script för att lägga upp data på hemsidan

I mappen `scripts` finns tre pythonscript. Ett som genererar alla lag, grupper och matcher. Ett som laddar upp nuvarande ställning i säsongen och ett som laddar upp historiska resultat. Dessa script läser in data från exceldokument som ligger i mappen `scripts/datasheets`.

För att få scriptet att fungera behöver man en service key som används för att få tillgång till databasen. Denna ska ligga i samma mapp som scripten.

2 Generera spelschema

Scriptet för att generera spelschemat till databasen rensar hela databasen innan den börjar. Exceldokumentet som ska redigeras ligger är `scripts/datasheets/schedule.xlsx`

1. Fyll i datum i cellen A1 på formen YYYY-MM-DD

2. Fyll i lagnamn i grupperna
3. Sätt tider i cellerna A4 och A6. Det går också att manuellt ändra tider på specifik match.
4. Sätt vilken plan matcherna spelas på i cellerna ovanför resultatcellen på varje match.
5. Sätt tider på sidan för slutspel på cellerna B1, B3 eller manuellt på matcherna.
6. Sätt vilken plan slutspelsmatcherna spelas på i cellerna ovanför cellen för lag 2 på varje match.
7. Se till att du har en service key till databasen i samma mapp som generateSchedule.py.
8. Kör generateSchedule.py så läggs all data upp till databasen.

3 Uppdatera säsongsställning

Efter ett delmästerskap i ÅM ska man uppdatera säsongsställningen i dokumentet scripts/datasheets/standings.xlsx.

1. Ändra namnet på denna säsong till årets säsong på cellen B4.
2. Tilldela varje sektion poäng från det gångna mästerskapet under rätt deltävling.
3. Se till att du har en service key till databasen i samma mapp som uploadStandings.py.
4. Kör uploadStandings.py så läggs allt upp på hemsidan.

4 Uppdatera historik

Efter en hel säsong har passerat ska man lägga till den gångna säsongen i de historiska resultaten. Detta görs i filen scripts/datasheets/history.xlsx

1. Scrolla ner i dokumentet tills det finns tomma rutor att fylla i.
2. Skriv in den nya säsongens topp-3 resultat på samma format som ovan. (En rads mellanrum mellan säsongerna krävs)
3. Se till att du har en service key till databasen i samma mapp som uploadHistory.py.
4. Kör uploadHistory.py så läggs allt upp på hemsidan.

5 Problem

Om problem uppstår som inte går att lösa på egen hand kontakta Jonas Teglund.