

Arbeidskrav 1- Feilsøking

I dette dokumentet forklarer jeg fremgangsmåten til hvordan jeg fant og rettet de seks feilene i koden i arbeidskravet.

1. Jeg åpnet filen i nettleser og sammenlignet med referansebildet.
2. Jeg gikk inn på Inspect og deretter i Console i nettleseren for å se etter feilmeldinger.
3. Jeg kjørte koden gjennom W3C Validator for å finne hva som var feil.
4. Jeg brukte kursmateriell og KI for å forstå og bekrefte løsningene.

Det var krevende i starten for jeg skjønnte ikke først hvor jeg skulle starte. Jeg begynte først å endre på HTML koden. På <article>- taggen begynte jeg å skrive inn navnene til brukerne der det sto «Navn Navnesen», siden jeg trodde jeg skulle fylle inn navnene deres inn der. Etter å ha lest pensum på nytt forsto jeg at på JavaScript så skal man ikke endre HTML koden.

Da jeg skjønnte fremgangsmåten, ble det enklere å gå frem. Jeg fant en del skrivefeil så jeg endret på dem. Jeg brukte verktøy som **Webtricks** for å gå gjennom tidligere pensum på hvordan man gjør feilsøking. Jeg brukte også en kode **validator** som viser hva som er feil med koden. Deretter gikk jeg inn på nettleseren og åpnet **Console** inne på Inspect for å se etter feilmeldinger.

Jeg valgte også å legge til en ekstra <article> for den tredje brukeren siden jeg så det manglet. Dette var i HTML koden så ble usikker på om det regnes som en feil, men jeg valgte å ta det med for sikkerhet skyld. Det hadde ingen praktisk betydning, resultatet ble det samme med og uten denne endringen.

Til slutt brukte jeg **KI** som støtteverktøy for å få en bedre forståelse av feilene og for å verifisere funnene mine.

Referanseliste

- Google. (2025). *Gemini* (22. september). [Stor språkmodell].
<https://g.co/gemini/share/1722ba25f8f7>
- Webtricks. (2025). *Feilsøking*. <https://lms.webtricks.blog/kurs/innforing-i-programmering/feilsoking/konsollen-viktigste-feilsokingsverktøyet>
- W3C. (2025). *Nu Html Checker*. <https://validator.w3.org/nu/#textarea>