

## Termin oppgave dokumentasjon:

### Uke 47

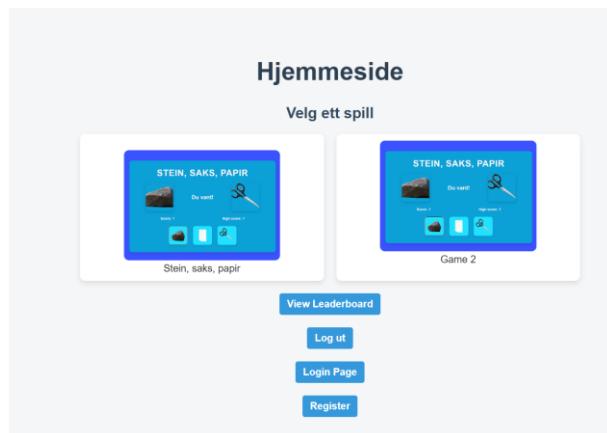
Jeg har planlagt hvordan og hva jeg skal gjøre med terminoppgaven.

Jeg skal lage en nettside der man logger inn for å spille noe, også se på leaderboard med andre spillere.

### Uke 49:

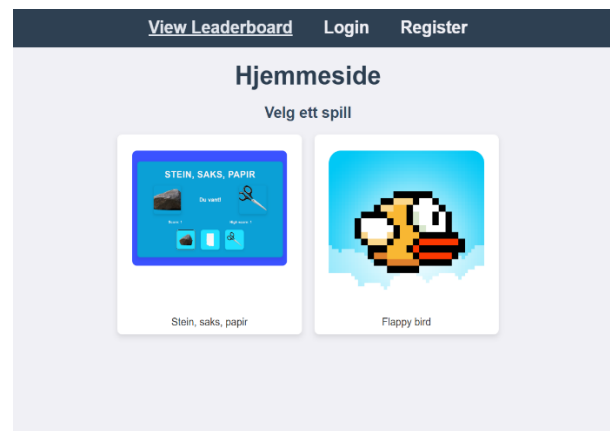
Jeg lager det med Flask Python.

Dette er hva jeg hadde:



Dette er stilingen jeg gjorde:

(ikke logget inn)



- Jeg lagde et helt logg inn system, som bruker 2fa for å kunne logge inn, og bruke innholdet.
- Jeg lagde et stein, saks, papir spill, på egen hånd, og satte det inn i nettsiden.
- Jeg har endret på stilingen på nettsiden, for at det skulle se bedre og mer proff ut.
- La til header, og fjernte linker som ikke skal vises når man er logget inn som f.eks login page, register.
- Jeg la til et spill fra Github, og gir kreditt til brukeren på spill siden (Flappy Bird).
- Jeg har lagt inn leaderboard til det spille jeg hadde fra før, og jobber nå med å legge til en leaderboard for github spillet.
- La til koden i GitHub

## Uke 50:

- Jeg brukte en del tid på å fikse leaderboard for flappy bird. Jeg tenkte at jeg kunne legge til scoren i samme class som den andre leaderboardet, men det fikk jeg ikke til, så jeg lagde en annen en.

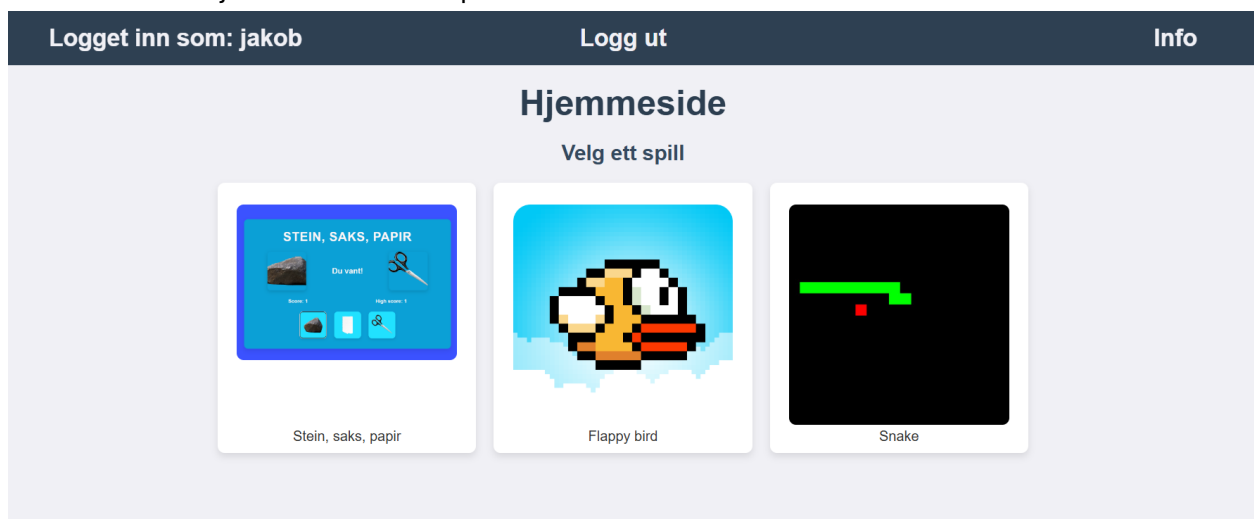
```
class FlappyHighScore(db.Model):
    id = db.Column(db.Integer, primary_key=True)
    user_id = db.Column(db.Integer, db.ForeignKey('user.id'), nullable=False)
    flappyScore = db.Column(db.Integer, nullable=False)
    user = db.relationship('User', backref=db.backref('flappy_high_scores', lazy=True))

class SnakeHighScore(db.Model):
    id = db.Column(db.Integer, primary_key=True)
    user_id = db.Column(db.Integer, db.ForeignKey('user.id'), nullable=False)
    snakeScore = db.Column(db.Integer, nullable=False)
    user = db.relationship('User', backref=db.backref('snake_high_scores', lazy=True))
```

Etter jeg gjorde det, og justerte på koden, så funka det.

- Jeg la til en info side, der det står hva nettsiden er, og hvordan man bruker det, i form av en brukerveiledning.
- Jeg la til det nye på GitHub, også klonet jeg det til ubuntu server. Da satte jeg opp en virtual environment, men Flask og Gunicorn, også hostet jeg nettsiden med gunicorn. IP adresse: 10.2.3.91:8000
- Nettsiden gikk veldig sakte, så jeg la til gunicorn\_config.py filen som ble sendt på teams, hostet med den, også gikk det veldig godt.
- Jeg la til et Snake spill som jeg hentet fra GitHub, men det var ikke helt som jeg ønsket, så jeg justerte en del på koden for at det skulle være bra. Jeg la til ca 50, ekstra linjer med kode, av nå 150 linjer.
- Jeg la til leaderboard til snake. Dette gjorde jeg ganske lett etter å ha gjort det 2 ganger.
- Jeg har legget inn at man kan se om man er logget inn, og hvilken bruker man er logget inn i, i header.
- Jeg satte alt inni github igjen, og pulla det på serveren, hvor den er hostet igjen på 10.2.3.91:8000.

Dette er hvordan hjemmesiden ser ut på slutten:



## Ip

<b>Ubuntu server ip</b>
10.2.3.91

Ble ikke feridg med:

Jeg ville legge til en søkefelt for å kunne åpne spill ved å søke på det, men jeg rakk ikke å gjøre det fordi jeg måtte sette andre ting over det. Et søkefelt hadde ikke vært så bra på denne nettsiden uansett.

Ellers er jeg fornøyd med det jeg lagde.