



id: SI_002

model: openai/gpt-5.4-mini

seeds:

04_risiko: sårbarhet

03_relasjon: tillit

05_handling: utforsk

01_perspektiv: operasjonell

02_tidhorisont: langsiktig

Instans SI_002

Seeds: operasjonell | langsiktig | tillit | sårbarhet | utforsk

Den viktigste blindsonen er å tro at dette først og fremst er et **innholdsproblem**. Med bare 10–12 emner over tre år er det egentlig et **kapasitets- og tillitsproblem**: Hva skal studentene kunne gjøre sikkert etter 3 år, og hva må de kunne utforske selv senere?

1) Operasjonell blindsone: for mange emner, for lite integrasjon

Et studieprogram kan lett ende med 11 “små siloer” der hver faglærer optimaliserer sitt fag. Resultatet blir bredde på papiret, men svak systemforståelse i praksis. Et viktig spørsmål er derfor: **Er emnene koblet til en tydelig progresjon i ferdigheter?**

Anbefaling: legg inn færre, større byggesteiner med tydelig nivådeling:

- **Grunnlag:** matematikk, diskret matematikk, programmering
- **System:** databaser, operativsystem/nettverk, arkitektur
- **Produksjon:** software engineering, test, sikkerhet, DevOps/MLOps
- **Anvendelse:** AI-verktøy og ett integrert prosjektforløp

2) Langsiktig blindsone: dere utdanner for et fag som endrer seg raskere enn emneplanen

Å undervise “riktige verktøy” er mindre verdifullt enn å bygge **lære-å-lære-kompetanse**. En kandidat som kan vurdere dokumentasjon, bruke API-er, teste hypoteser og lese kode, er mer robust enn en som kun kjenner dagens rammeverk.



Vei mot hverandre:

- **Bredde-argumentet:** matematikk/fysikk gir dypere modellforståelse og evne til å lære nye teknologier.
- **Spesialiserings-argumentet:** næringslivet etterspør umiddelbar produktivitet i software og AI.
- **Mitt avvikende poeng:** Hvis spesialiseringen blir for tidlig, svekkes den langsiktige læringsevnen. Legg spesialisering sent, ikke tidlig.

3) Tillit og sårbarhet: industrinærhet kan bli en styrke og en risiko

Nærhet til forsvarsindustri og høyteknologi gir legitimitet, praksisprosjekter og relevans. Men det kan også skape **for sterk programdrevne forventninger fra arbeidsgivere**. Da risikerer man å gjøre bacheloren til en opplæringskanal for lokale behov, ikke et universitetsprogram.

Blindsoner:

- Å undervurdere **etikk, sikkerhet og ansvarlig AI**
- Å anta at “industrien vet best”
- Å glemme at studenter trenger trygghet til å feile i kontrollerte rammer

4) Hva bør prioriteres?

Jeg ville prioritert:

1. **Kjerne i programmering + systemforståelse**
2. **Sterk SE-praksis:** testing, versjonskontroll, arkitektur, samarbeid
3. **AI-verktøy som ferdighet, ikke som eget stort fag**
4. **Et integrert design-/capstone-emne** med reelle krav

Neste steg

1. Kartlegg hvilke læringsutbytter som er “må ha” ved exit etter 3 år.
2. Slå sammen overlappende emner og lag en progresjonsmatrise fra 1. til 6. semester.
3. Etabler et rådgivende panel med både industri, faglærere og studenter for å teste om planen gir tillit i praksis.