

RISIKO ANALYSE AF MARIO'S PIZZA

VURDERING AF HVILKE RISICI DER ER:

Præsentation med risikoanalyse og risikoplan over projektet

- Sygdom forhindrer samarbejde
- Fejl i kode (fejl i test)
- Fejlkommunikation ift. opgaven
- Projektet bliver ikke gemt på computeren
- Systemnedbrud
- Tidsbegrænsning forhindrer i blive færdig til tiden
- Uenigheder og konflikter i gruppen

EVENTUELT:

- Præciserer vores risici, så det er lettere at forstå
- Overveje, om vi har lyst til at gå mere i dybden med vores ricisi
- Fejl i kode: (Er der fejlkommunikation (fejl i krav) eller er det fejl i koden selv)

VURDERING AF HVOR SLEMME RISICIENE ER (1-5) OG KONSEKVENSEN (1-10):

- 1. (3 - 3) Sygdom der forhindrer vi kan arbejde sammen
- 2. (4 - 7) Fejl i kode
- 3. (2 - 10) Fejlkommunikation ift. opgaven
- 4. (1 - 10) Projektet bliver ikke gemt på computeren
- 5. (3 - 6) Systemnedbrud
- 6. (3 - 10) Tidsbegrænsning forhindrer at blive at færdig til tiden
- 7. (2 - 5) Uenigheder og konflikter i gruppen

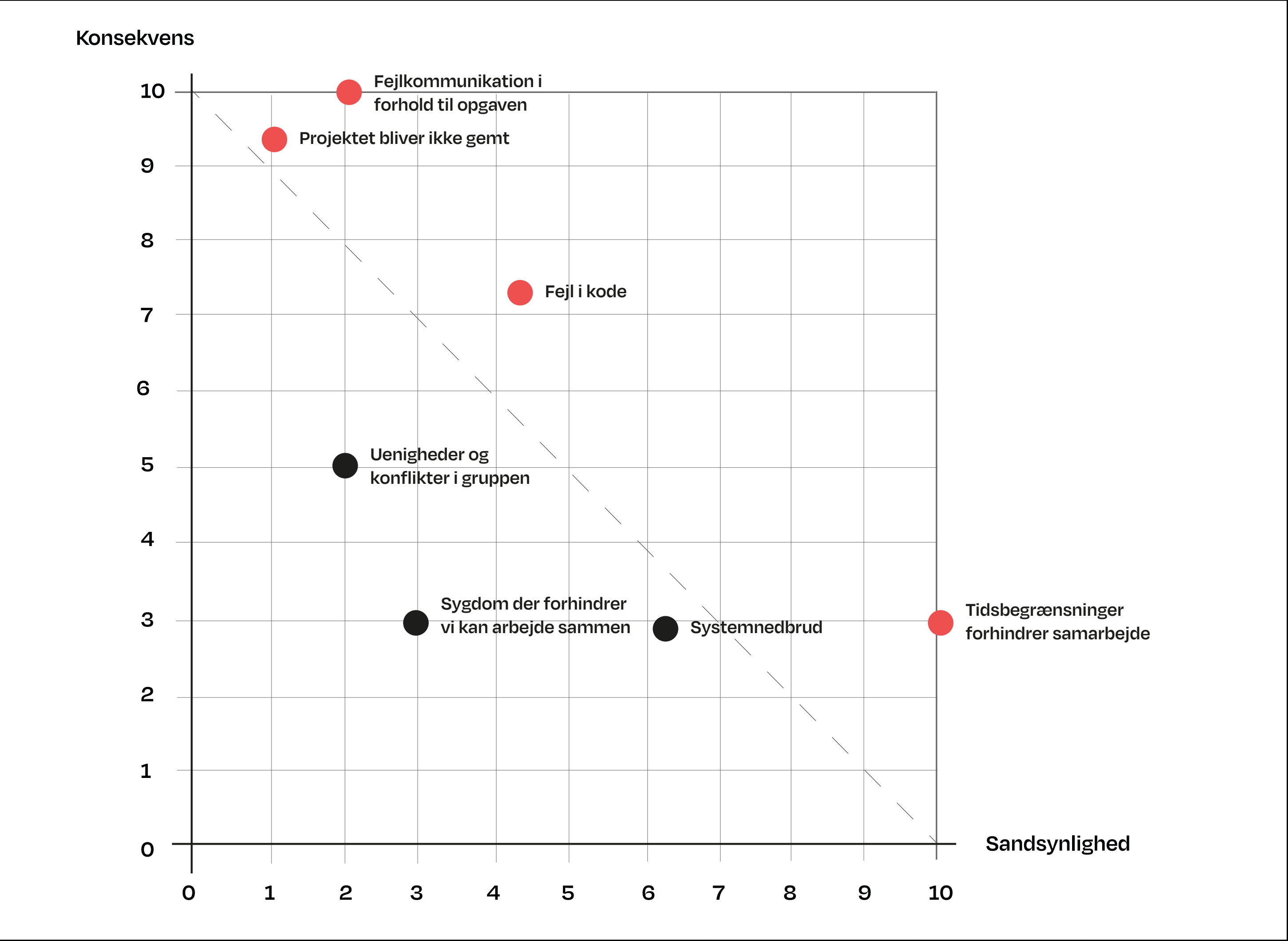
PRIORITERING AF RISICIENE:

- 1. Tidsbegrænsninger forhindrer samarbejde
- 2. Fejl i kode
- 3. Fejlkommunikation i forhold til opgaven
- 4. Systemnedbrud
- 5. Projektet bliver ikke gemt
- 6. Uenigheder og konflikter i gruppen
- 7. Sygdom der forhindrer vi kan arbejde sammen

RISIKO ANALYSE:

| RISIKOMOMENT | SANDSYNLIGHED | KONSEKVENNS | PRODUKT |
|---|---------------|-------------|---------|
| <i>SYGDOM DER FORHINDRER SAMARBEJDE</i> | 3 | 3 | 9 |
| <i>FEJL I KODE</i> | 4 | 7 | 28 |
| <i>FEJLKOMMUNIKATION I FORHOLD TIL OPGAVEN</i> | 2 | 10 | 20 |
| <i>PROJEKTET BLIVER IKKE GEMT</i> | 1 | 10 | 10 |
| <i>SYSTEMNEDBRUD</i> | 3 | 6 | 18 |
| <i>TIDSBEGRÆNSNINGER FORHINDRER AT BLIVE FÆRDIG TIL TIDEN</i> | 3 | 10 | 30 |
| <i>UENIGHEDER OG KONFLIKTER I GRUPPEN</i> | 2 | 5 | 10 |

RISK MANAGEMENT PLAN:



UDVIKLING AF EN PLAN FOR RISIKOHÅNDTERING:

TIDSBEGRÆNSNING, DER FORHINDRER I AT BLIVE FÆRDIG TIL TIDEN:

- For at undgå at vi er tidsbegrænset, så prøver vi at arbejde på forkant ved hjælp af Scrum. Dette gør vi ved at holde hinanden opdateret, ved at holde Daily Sprint Meeting, så vi alle er på samme bølgelængde og har det samme overblik med opgaven. Dette møde kan vi eventuelt samle op på mulige fejl, så vi kan håndtere dem og undgå dem.

SYGDOM DER FORHINDRER VI KAN ARBEJDE SAMMEN:

- Ved at arbejde på forkant med Daily Scrum Meeting, og uploading af GitHub, så er vi også på forkant hvis der opstår sygdom. Vi sikrer os nemlig, at vi alle er på lige fod og har hinandens dokumenter og koder, så vi ikke er afhængige af hinanden.

UENIGHEDER OG KONFLIKTER:

- For at undgå at vores uenigheder og konflikter eskalerer, så har vi mulighed for at snakke ud om det til vores Daily Scrum Meeting. Det hjælper vores Scrum Master Peyton, der sørger for at vi kan fortsætte arbejdet ved at håndtere uenighederne.

FEJLKOMMUNIKATION I FORHOLD TIL OPGAVEN

- Ved benyttelse og vejledning af vores lærere, så har vi rig mulighed for at sikre os at vi har forstået opgaven på den rigtige måde. Gennem vejledning og kommunikation, så kan vi løbende forhindre misforståelser.

FEJL I KODE // SYSTEMNEDBRUD // PROJEKTET BLIVER IKKE GEMT:

- Ved hjælp af GitHub, så kan vi nemt uploade programmeringerne, så vi alle kan tjekke op på hinandens koder, hvis nu der skulle opstå problemer med fejl i ens kode eller at projektet ikke er gemt. Dermed har vi mulighed for at kunne følge med hurtigt igen, hvis nu at intet er blevet gemt på ens computer og/ eller computeren går ned.