

How to - Daggi

Innledning

- **Om oss**
Kort om hvem gruppa
- **Om prosessen**
Tankegangen om prosjektet fra vi startet til vi var "ferdig"
- **Om arbeidet**
Arbeidsfordeling. Arbeidsstruktur
- **Arbeidsmetode**
Eierskap, tempo, utviklingsmiljø og samarbeidsplattform
- **Om koden**
Kodeutvikling/arbeidsmetode
- **Forutsetninger for oppgaven**
Hvordan oppgaven skal være blandet mellom dokumentasjon og kode?

Problemstilling

- **Problemstilling:**
Beskriv hvorfor man trenger vårt produkt. Problemer man møter på uten vårt produkt. Hva løser vårt produkt
- Plan for løsning:
Hvordan løse problemet. Kort om hvordan det ser ut. Hva vi lager som gir mest verdi
- **Begrensninger og antakelser**
Hva det er tid til å lage. Hva er nødvendig å ta med. Se på begrensninger med de kravene vi tar med. Hva krav vi ikke får tatt med. Hva kortsiktige løsninger vi har valgt
- **Krav til begrensninger**
Hva vi ikke fikk/ville få tid til å gjøre som kunne ha vært viktig
- **Begrensninger av kode**
Hva vi ikke fikk gjort som f.eks. systemtester. Ting som har vært et hinder, kanskje intellij eller tiden vi fikk på oppgaven

Hoveddel

- **Metode**

Hvordan vi fant krav og kunnskap om hva problemet produktet skal løse.
F.eks. Use-case osv...

- MVP
Hvordan vi forholdt oss til MVP (minimum viable product)
- **Funksjonelle og ikke-funksjonelle**
Forklare funksjonelle og ikke-funksjonelle krav. Og at (ikke hvordan) implementasjon av dem påvirker koden
- Personas
Lage flere realistiske brukere og hvordan vårt produkt skal hjelpe han/hun
- **Use case**
Vise til use-case diagrammer som beskriver interaksjonen mellom person og systemet. Ved start av prosjektet og ved slutt
- Use case beskrivelse
Beskrivelse av hva diagrammene dekker
- **Ikke funksjonelle krav**
Beskrive de ikke funksjonelle kravene (hvis de finnes).
- **Estimering og planlegging**
Beskrive hva det går ut på. Og hvordan vi gjorde det. F.eks. T-shirt sizing
- Bruker
Hva er bruker skal kunne gjøre
- Selger/eier
Hva en selger skal kunne gjøre
- Admin
Beskrive hva en eventuelt admin kan gjøre
- Bruker
- Selger/eier
- Admin
Om noe blir endrer skriv det på hver av de. Hvor og hvorfor, og hvordan det påvirker bruker, selger eller/og admin
- **Systemkrav/spesifikke krav**
Beskriv metoder som dekker kravene. f.eks. KjøpBillett() + beskrivelse av hva krav den dekker og kort om hvordan det skjer.
- **TDD/Tester**
Hva utviklingsmetode og testdreven utvikling vi tok i bruk
- **Applikasjon og Arkitektur**
Diagram av noe slag som viser arkitekturen. Med en liten beskrivelse av hva man ser. F.eks. hva "tykk pil" er
- Parkeringsplass (klassen)
Beskrivelse av klassen. Hva den implementerer. Har den en superklasse osv..
- (En annen klasse)
- (En annen klasse)
- ...
- **Sekvensdiagram**
Sekvensdiagram av koden under og etter koden er ferdig. Sammen med en beskrivelse
- **Sekvens og Aktivitetsdiagram, for hver person**
- Bruker1 (eks Berg Tore Marcusen)

Det blir beskrevet et scenario. Et krav skal dekke scenarioet. Vist med både aktivitetsdiagram og sekvensdiagram

- Bruker 2
- Bruker 3

..

- **Inkludert og ekskludert, hva og hvorfor**

Beskriv hva vi har utelatt og hvorfor

- **Svakheter ved systemet**

- Parkeringsplass har ikke kart (f.eks.)

Beskriv hvorfor parkeringsplass ikke har kart og hvorfor dette er en svakhet

- ..

System Forklaring

- **Hvordan kjører prototypen**

Beskrivelse med bilder og tekst hvordan man bruker prototypen

Konklusjon

Beskrive vår opplevelse av prosjektet. hva som var bra, hva som var dårlig.

Hva som gjorde ting lettere eller vanskeligere for oss.

Hva vi lærte

Og hva vi kan fortsette på ved eventuelt videreutvikling