

UKE 8 - Håndtering av krav

IN1030 - Gruppe 1 & 3D

Plan for timen

Krav

Forstudie

Kravspesifikasjon

Validering av kravspesifikasjon

Kravendringer

Ukesoppgaver



OBLIG 4

Fristen for å melde seg til en gruppe i dag (18.03)

Oblig 4 blir lagt ut på onsdag (23.03)

Gruppearbeid - gode linker:

For samarbeid: Zoom, Teams, MetroRetro, Google Drive.

For tegning av UML:

<https://www.draw.io>

Læringsmål - forklart:

Kan du anvende metoder og teknikker for **kravhåndtering**:

- innhente, analysere og spesifisere krav.
- skille mellom funksjonelle og ikke-funksjonelle krav.
- skrive kravspesifikasjon.
- utføre modellering av UML-diagram basert på krav (neste uke).

Hva er krav?

Vi har ulike **typer** krav: *funksjonelle krav og ikke-funksjonelle krav*
(produktkrav, organisatoriske krav og eksterne krav)

Hvorfor?

Vi utvikler IT-systemer for å løse et problem/identifisere og utnytte muligheter.

Kravene forteller oss noe om hva som skal lages.

Kostbart å rette feil i kravene etter systemleveranse. Utilstrekkelig kravhåndtering å viktigste årsak for problemer i systemutviklingsprosjekter. Kravendringer vil alltid forekomme.

Kravene bør være:

- **Forståelige**: alle interessenter/stakeholders må kunne forstå kravspesifikasjonen
- **Testbare**: vi må kunne avgjøre om det ferdige systemet gjør det det skal
- **Sporbare**: vi må vite hvilken del av koden som skal endres når det kommer nye krav

Krav

Vi finner krav ved å analysere konteksten til et system. Og evt gjennom kontrakt med kunde. Krav bestemmer hvilke funksjoner et system skal ha.

Brukerkrav: påstander om planlagte tjenester og/eller begrensninger til systemet, basert på kunden/brukers behov. Fra brede beskrivelser av systemets egenskaper til detaljerte funksjonsbeskrivelser.

Systemkrav: detaljert, formell beskrivelse av programvarens funksjoner, tjenester eller operasjonelle begrensninger. Implementasjonen av disse skrives i detalj i et systemkrav dokument.

- Forskjellige typer krav er nødvendig for å kommunisere informasjon om systemet til forskjellige lesere.
- Man trenger å skrive kravene på forskjellige nivåer av hvor detaljert de er fordi forskjellige typer av lesere eller mottakere bruker de på forskjellige måter
- Brukerkrav
 - Ofte skrevet i et naturlig og gjenkjennelig språk for de aller fleste
 - Hvilke funksjoner systemet er forventet å tilby til brukerne
 - Brukerkrav er ofte mer generelle
 - De som har behov for brukerkravene, de har ofte ikke et behov for å vite hvordan systemet skal bli implementert.
- Systemkrav
 - Skal definere nøyaktig hva som skal implementeres

Funksjonelle krav

Hva systemet skal gjøre (men kan også beskrive hva systemet ikke skal gjøre):

- Hvilke tjenester/funksjoner skal systemet tilbyr?
- Hvordan skal det reagere på ulike typer input?
- Avhenger av hvilket system som skal utvikles, systemet brukere og de som er ansvarlig for å beskrive kravene
- Variere fra generelle krav til hva systemet skal gjøre, til mer spesifikke krav som reflekterer arbeidsmetoder eller en organisasjons allerede eksisterende system
- Funksjonelle krav skrives gjerne på formen «systemet skal... /systemet bør (nice to have)...»

Funksjonelt brukerkrav: En bruker skal kunne laste opp bilder.

Funksjonelt systemkrav: Systemet skal lagre bilder.

Ikke-funksjonelle krav

Krav som ikke direkte beskriver funksjonene som leveres til sluttbruker, men karakteristikk med systemet som helhet;
Hvordan skal det fungere?

Sier noe om **kvalitetsattributter** systemet skal ha

Sier noe om **egenskaper** - hvordan systemet skal oppføre seg

Bør være **målbare** (mulighet for å teste dem)

Sier noe om:

- Implementasjon
- Reliabilitet
- Reponstid
- Sikkerhet

Mer kritiske → Møter man ikke disse kan det føre til at hele systemet ikke fungerer.

Egenskap	Variabel (måling)
Hastighet	Antall transaksjoner/sekund Responstid Tid på oppdatering av skjermen
Størrelse	Gigabytes, use case-poeng
Enkelhet i bruk	Opplæringstid Antall hjelpebilder
Pålitelighet	Gjennomsnittlig tid til feil Sannsynlighet for utilgjengelighet Feilrate
Robusthet	Tid til oppstart etter feil % handlinger som fører til feil Sannsynlighet for ødelagte data ved feil
Flyttbarhet (portability)	% installasjonsavhengige kommandoer/setninger

Eksempler fra Ruter billettapp

Ikke-funksjonelle krav:

- En ny kunde skal kunne betale for en billett på under tre minutter
- Systemet skal kunne håndtere 10.000 brukere samtidig
- Systemet skal utvikles ved hjelp av smidige utviklingsmetoder

Funksjonelle krav:

- Systemet skal kunne vise en oversikt over en brukers betalte billetter
- Systemet skal gi en beskjed når det er under 24 timer til en billett løper ut
- Systemet skal tilby funksjonalitet for valg av billettype
- Systemet bør tilby hurtigkjøp av tidligere valgte billetter
- Systemet skal fjerne et valgt bankkort hvis brukere taster feil pin 3 ganger

Ikke-funksjonelle krav

Produktkrav: Brukskvalitet/brukervennlighet, ytelse og effektivitet samt lagringsplass, pålitelighet og lagring av data.

Eks: Systemet skal kun bruke 5 MB lagringplass på bilder.

Organisasjonskrav: kostnader & ressurser, leveransetidspunkt & prosess; utviklingsmodeller, programmeringsspråk, verktøy og komponenter samt generelle standarder og regler.

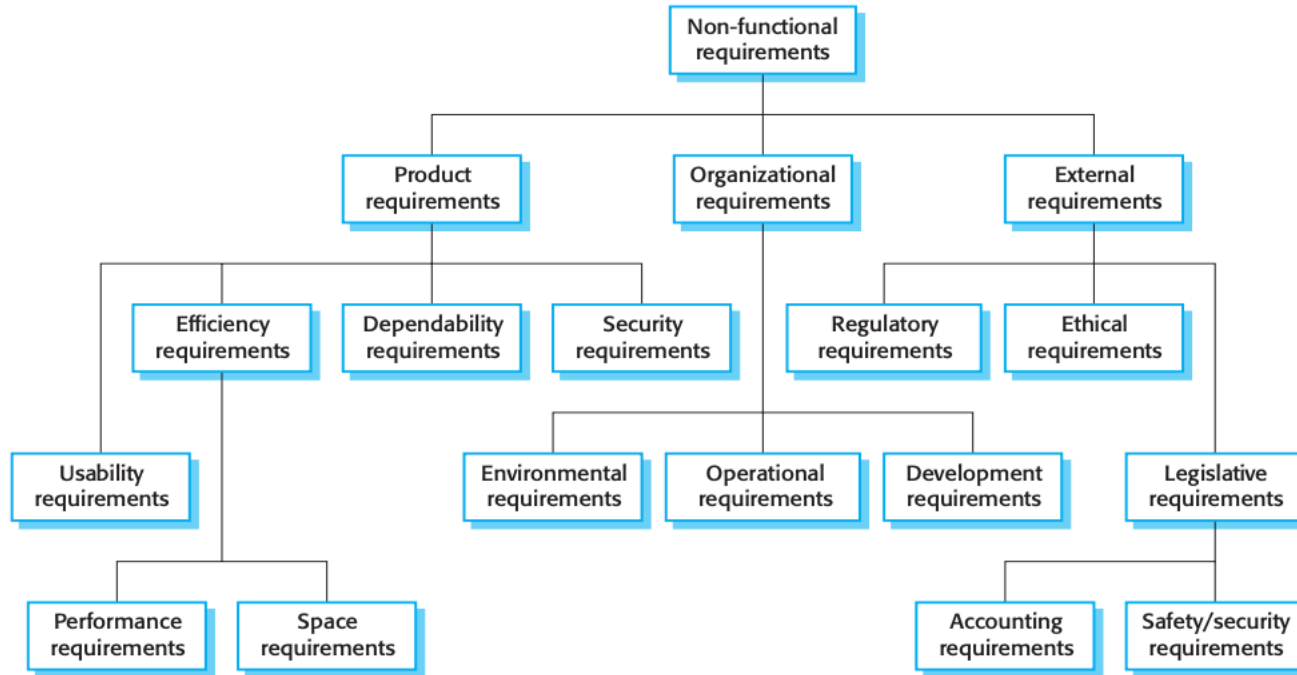
Eks: Systemets ulike funksjoner skal testes hver andre måned.

Systemets kodehåndtering skal gjøres i GitHub.

Eksterne krav: lovverk, begrensninger & etiske problemstillinger.

Eks: Systemet skal tilfredsstillte 39 av 61 WCAG-retningslinjer.

Typer av ikke-funksjonelle krav



Veien mot kravspesifikasjon

Hva er en kravspesifikasjon?

Et dokument som spesifiserer kravene til et system.

Hensikt:

“Basis for anbud - ulike tilbydere vil kunne tilby ulike måter å løse kundens behov på

Basis for kontrakt

Basis for design og implementasjon av systemet”

Dag Sjøberg

Kravhåndteringsprosessen

Steg 1: Forstudie/målanalyse

- Kost/nytte-analyser | Risikoanalyser | Gevinstrealisering

Steg 2: Kravinnsamling og kravanalyse

- Hva ønsker interessentene seg? Hva har de behov for?
- Prioritering av kravene

Steg 3: Kravspesifisering

- Utgangspunkt for anbud og kontrakt (mellom kunde og leverandør)
- Utgangspunkt for design, implementasjon og testing
- Utgangspunkt for estimater (tid og kostnad)

Steg 4: Validering av kravspec

Steg 5: Håndtering av kravendringer

Forstudie :

Hvem er aktør?

Hvilke funksjoner ønsker aktør?

Politiske, økonomiske eller alternative aktørers krav?

Ressursbruk: tid, kostnader, arbeidskraft.

Klassifisering, prioritering, dokumentasjon

Kravhåndteringsprosessen

Steg 1: Forstudie/målanalyse

- Kost/nytte-analyser | Risikoanalyser | Gevinstrealisering

Steg 2: Kravinnsamling og kravanalyse

- Hva ønsker interessentene seg? Hva har de behov for?
- Prioritering av kravene

Steg 3: Kravspesifisering

- Utgangspunkt for tilbud og kontrakt (mellom kunde og leverandør)
- Utgangspunkt for design, implementasjon og testing
- Utgangspunkt for estimater (tid og kostnad)

Steg 4: Validering av kravspec

Steg 5: Håndtering av kravendringer

Kravspesifikasjon :

Skaper felles forståelse av systemet.

Skaper enighet om hva som skal leveres.

Er grunnlag for kontrakt som viser hva leverandør og kunde blir enige om.

Forhindrer eventuelle konflikter som kan oppstå på bakgrunn av uklare forventninger.

Kravhåndteringsprosessen

Steg 1: Forstudie/målanalyse

- Kost/nytte-analyser | Risikoanalyser | Gevinstrealisering

Steg 2: Kravinnsamling og kravanalyse

- Hva ønsker interessentene seg? Hva har de behov for?
- Prioritering av kravene

Steg 3: Kravspesifisering

- Utgangspunkt for tilbud og kontrakt (mellom kunde og leverandør)
- Utgangspunkt for design, implementasjon og testing
- Utgangspunkt for estimater (tid og kostnad)

Steg 4: Validering av kravspec

Steg 5: Håndtering av kravendringer

Validering av kravspec :

Møter vi aktørers faktiske behov?

Finne eventuelle problem/mangler med kravspesifikasjonen.

Kravhåndteringsprosessen

Steg 1: Forstudie/målanalyse

- Kost/nytte-analyser | Risikoanalyser | Gevinstrealisering

Steg 2: Kravinnsamling og kravanalyse

- Hva ønsker interessentene seg? Hva har de behov for?
- Prioritering av kravene

Steg 3: Kravspesifisering

- Utgangspunkt for tilbud og kontrakt (mellom kunde og leverandør)
- Utgangspunkt for design, implementasjon og testing
- Utgangspunkt for estimater (tid og kostnad)

Steg 4: Validering av kravspec

Steg 5: Håndtering av kravendringer

Kravendringer :

Når forståelsen av et problem endrer seg.

Plan for kravhåndtering

Håndtering av kravendringer

VIDEO:

Planleggingsmøte med umulig utgangspunkt og hvor krav endres hele tiden

Ukesoppgaver

Denne uken

Kapittel 4: Requirements Engineering.

Neste uke - Smidige praksisser og teamarbeid

Husk å besvar nettskjema om gruppeinndeling innen fredag 18/3, kl. 23.59.

Oblig 4 publiseres 23/3. Deadline 13/4, kl. 23.59

jehank@ifi.uio.no

jamilakm@ifi.uio.no