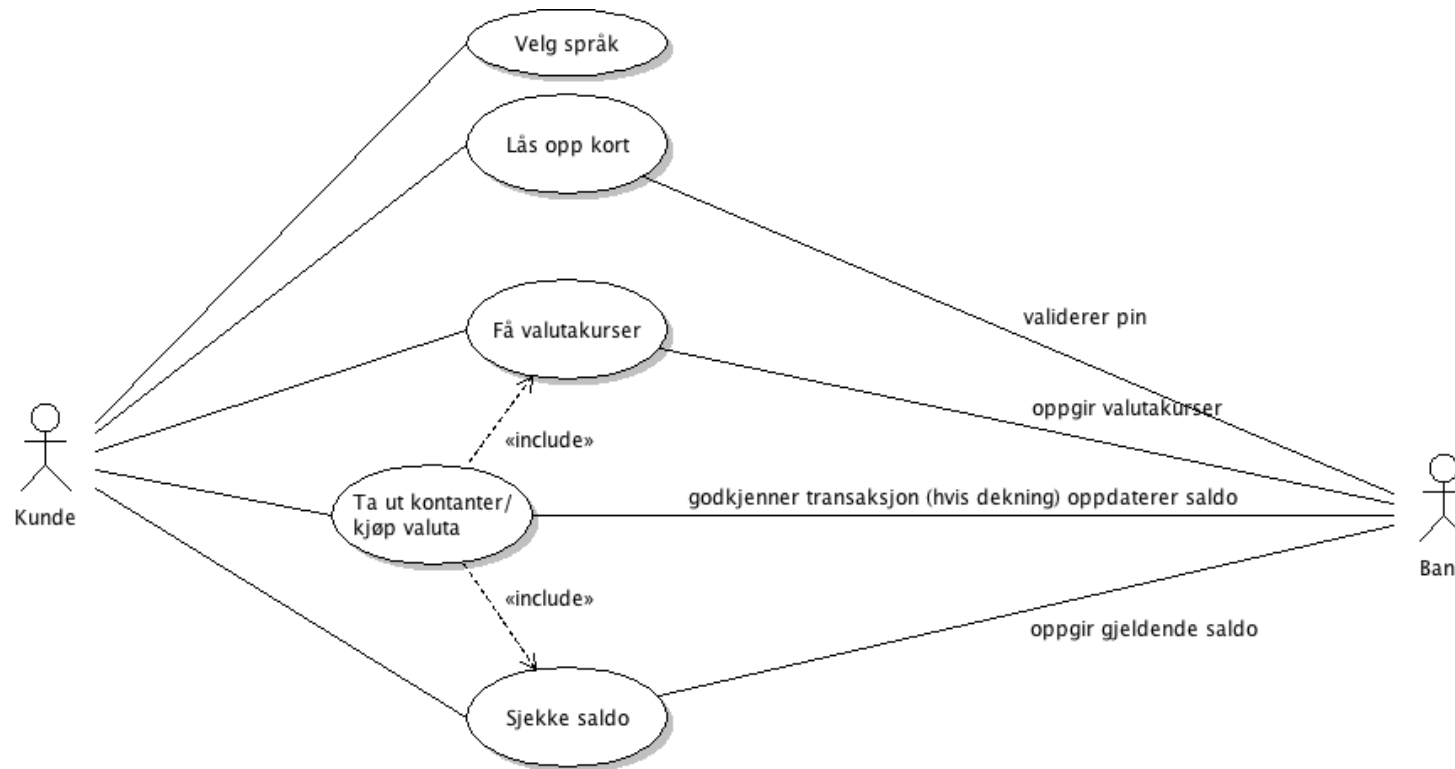


Minibank og billetterminal

Use case



Use case diagram

Jeg har tatt utgangspunkt i de funksjonelle kravene beskrevet på side 1 i oppgaveteksten (sjekke saldo, få valutakurser, ta ut kontanter). Jeg har tatt med ytterligere to funksjonelle krav:

-velg språk

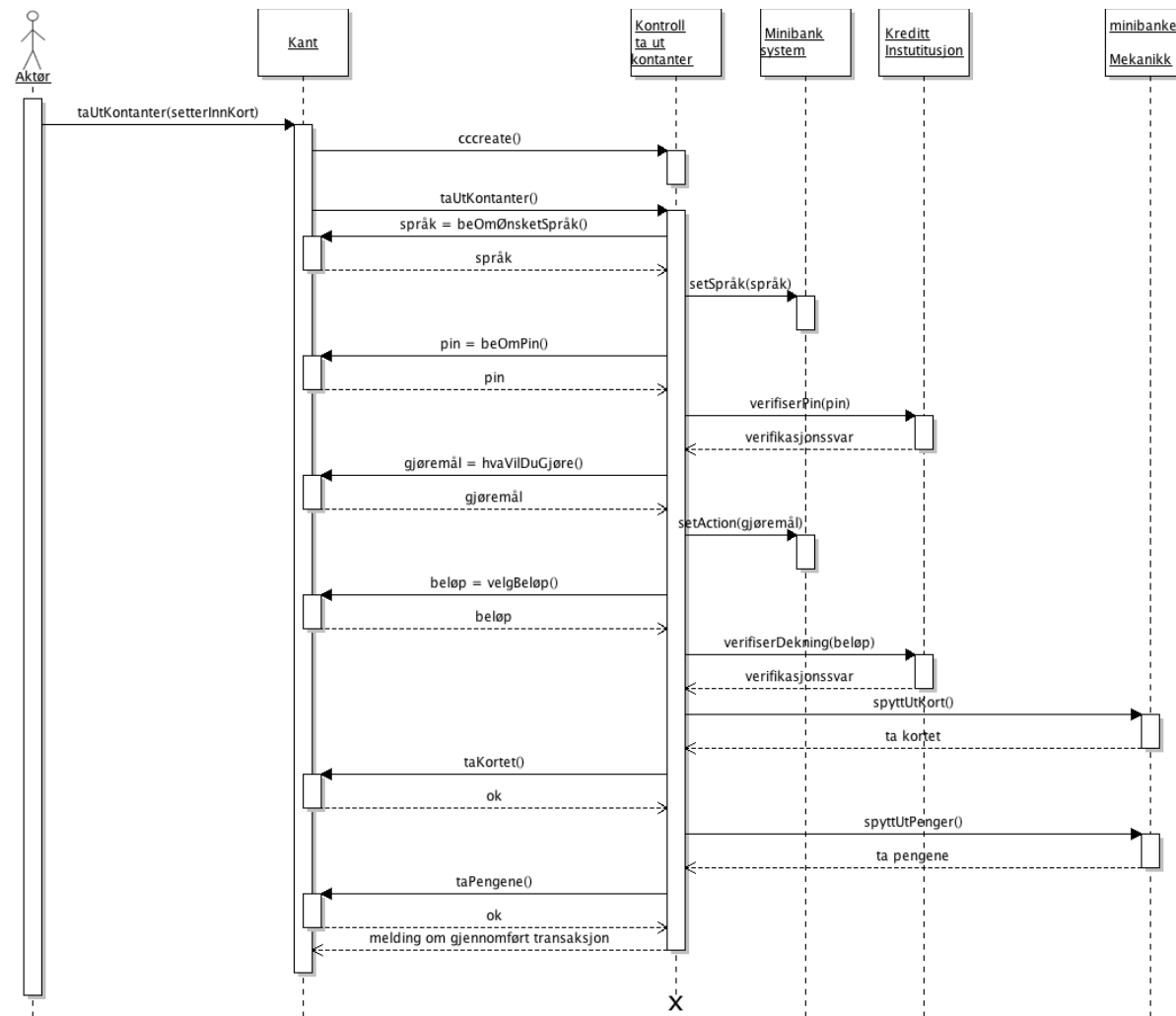
-lås opp kort (sikkerhet er et ikke-funksjonelt krav, men impliserer det funksjonelle kravet lås opp kort)

Jeg mener dette er et sentrale bruksmønster som alltid må til og alltid gjøres i sammenheng med de andre bruksmønstrene som oppgaven skisserer.

-jeg har slått sammen ta ut kontanter og kjøp valuta til et generelt bruksmønster der ønsket valuta er et valg.

-for å ta ut kontanter eller kjøpe valuta så må maskinen sjekke saldo (for dekning) og valutakurser(hvis et annet lands valuta er ønsket) og derfor inkluderer dette kall til bruksmønstrene 'Sjekk saldo' og 'Få valutakurser'.

Sekvensdiagram



Sekvensdiagram

Hovedflyten til 'ta ut kontanter' baserer seg på:

- aktør har gyldig kort
- aktør har gyldig pinkode
- aktør vil ta ut kontanter i aktuelt lands (det landet minibanken befinner seg) valuta så ingen valutaomregning er nødvendig.
- beskrivelse av lovlige beløp for uttak vises på skjermen og aktør følger dette.

Ikke-funksjonelle krav

Effektivitetskrav

(Ytelseskrav)

- Det skal i 90% av tilfellene ikke ta mer enn 3 sekunder fra man har skannet strekkoden til informasjonen er tilgjengelig og aldri mer enn 10 sekunder. Evalueres av eksternt testteam med automatiserte systemtester i testmiljø.

Pålitelighetskrav

- Oppetiden skal være på 99.8%. Monitoreres over tid og evalueres av utviklerne.

Pålitelighetskrav

- Skulle eksterne informasjonskilder gå ned (feks leverandøren av metrologisk informasjon), så skal resten av systemet ikke berøres. Evalueres med eksternt testteam og brukerundersøkelser i lab.

Sikkerhetskrav

- Sensitiv informasjon skal og all passasjerinfo skal krypteres i henhold til gjeldende industristandard. Evalueres av systemarkitekt og testteam med automatiserte systemtester i testmiljø.

Bruklarhetskrav

- Adgangskodene for internettilgang skal være personlige og enkle å huske. Evalueres med brukerundersøkelse i lab og spørreundersøkelse på nett.

Arkitektur

3-lags logisk arkitektur

- Presentasjonslag (telefon/tabletinterface)
- Forretningslogikklag
- Datalag

4-lags fysisk arkitektur

- Klient
- Webserver
- Applikasjonsserver
- Database

Testing

Enhetstesting

Her testes de individuelle programenhetene eller objektklassene. Metoder og objekter testes hver for seg. Dette skjer tidlig i utviklingsprosessen av den aktuelle enheten.

Integrasjonstesting

Her testes de elementene sammenesatt til større komponenter. Man tester det ofte igjennom grensesnittet til komponenter. Formålet er å teste at enhetene virker sammen på den måten det er tiltenkt. Testes når man mener en enhet er klar for å integreres i systemet.

Systemtesting

Noen eller flere komponenter er satt sammen til et system og dette systemet testes helhetlig. Man legger særlig vekt på å teste interaksjonen mellom komponentene. Formålet er å teste at komponentene virker sammen på den måten det er tiltenkt og at systemet fungerer helhetlig slik utviklerne har tenkt.

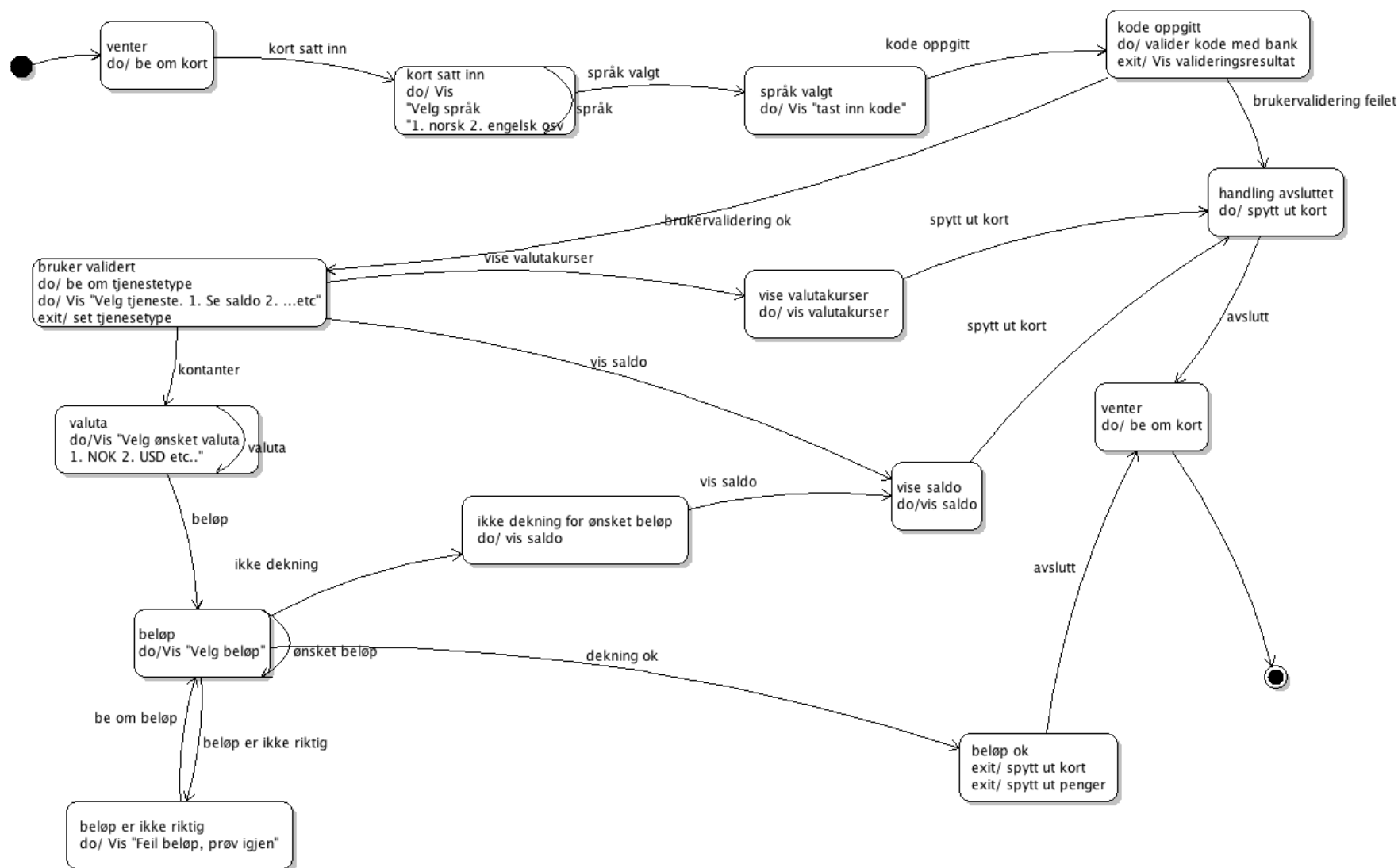
Akseptansetesting.

Her skal man evaluere systemet som leveres. Man vil verifisere at systemet som leveres er det kunden faktisk vil ha, at det tilfredsstiller kundens krav og at det er klart til bruk.

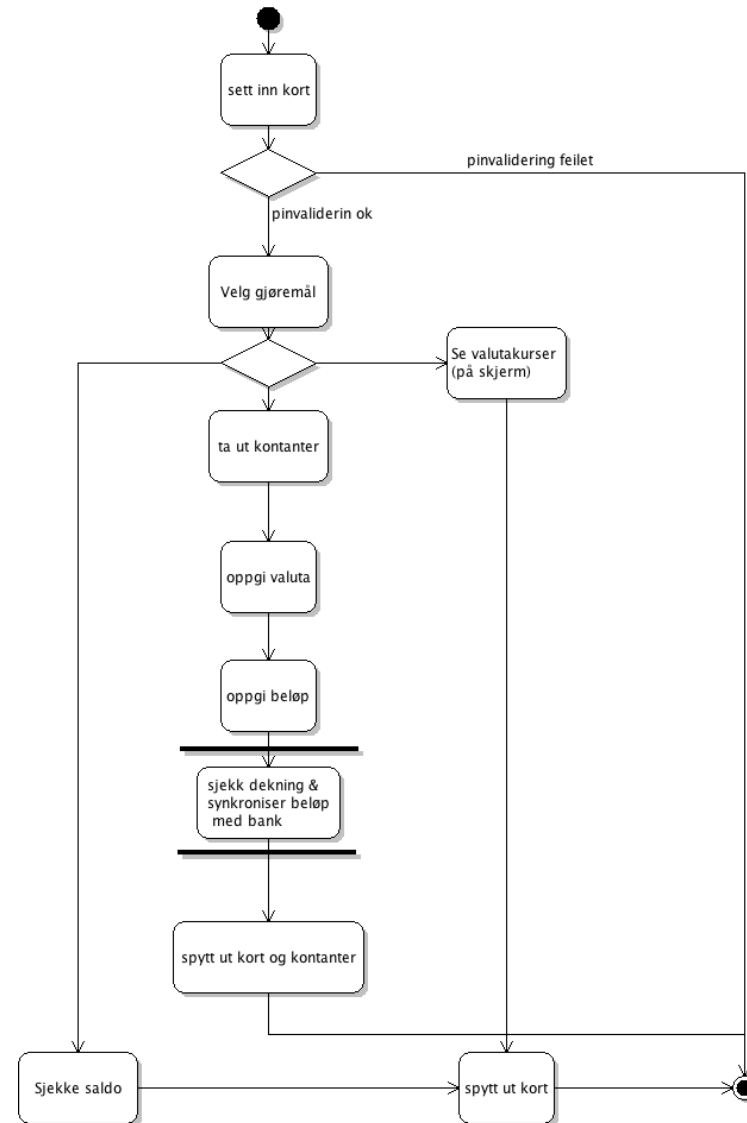
Forskjellen mellom enhetstesting, integrasjonstesting, systemtesting og akseptansetesting er at de ulike testene foregår på ulikt stadie i utviklingsprosessen. Hvor store deler av systemet som testes samtidig og hva som er formålet med testingen. Forskjellene mer spesifikt går frem av beskrivelsene i svar 4a

Den delen av billetterterminalen som henter informasjon tilknyttet passasjer, sete, flyrute samt mulighet for oppgradering av setetype må enhets-, integrasjon- og systemtestes. Delene som henter informasjon fra de ulike flyplassenes interne systemer (avgangstidspunkt, terminal, gate, adgangskode for internett, rabattkuponger og informasjon om VIP-lounge) må integrasjon- og systemtestes. Delene som henter informasjon fra NorStat (visum- og valutainformasjon, vind- og værforhold) må systemtestes. Systemet som helhet må akseptansetestes.

Tilstandsdiagram



Aktivitetsdiagram



Tilstands- og aktivitetsdiagrammer

Et tilstandsdiagram viser hvordan systemet reagerer på interne og eksterne hendelser. Det er spesielt egnet for å modellere "real time"-tilstander og viser hvilke muligheter som ligger i systemet i de enkelte tilstandene og hva som blir påvirket i de ulike tilstandene. For eksempel kan det være nyttig for å vise at en mikrobølgeovn ikke skal operere hvis døren lukkes opp.

Et aktivitetsdiagram viser dataflyten med fokus på aktivitetene i et gitt system. Det er spesielt nyttig for å vise hvordan bruksvalg endrer flyten i systemet og for å vise hvor i systemet datasynkronisering er viktig. Det kan for eksempel vise hvilke muligheter en bruker har i en nettbank ved overføring av penger til og fra ulike kontotyper og når det er nødvendig og synkronisere data i forbindelse med endring av saldo.