

Dokumentation

Semesterprojekt 3. Semester

Gruppe 10

Vejleder: Søren

Gruppemedlemmer:

Navn	Studienummer
Tonni Follmann	201504573
Stefan Nielsen	2222222
Mikkel Busk	33333
Halfdan	0
Ahmad	0
Jacob	0

Indhold

Indhold	i
1 Indledning	1
1.1 projektformulering	1
2 Kravspecifikation	2
2.1 Aktør-Kontekst	2
2.2 Use-Cases	3
2.3 Ikke-funktionelle krav	6
3 Hardware Arkitektur	7
4 Software Arkitektur	9

Kapitel 1

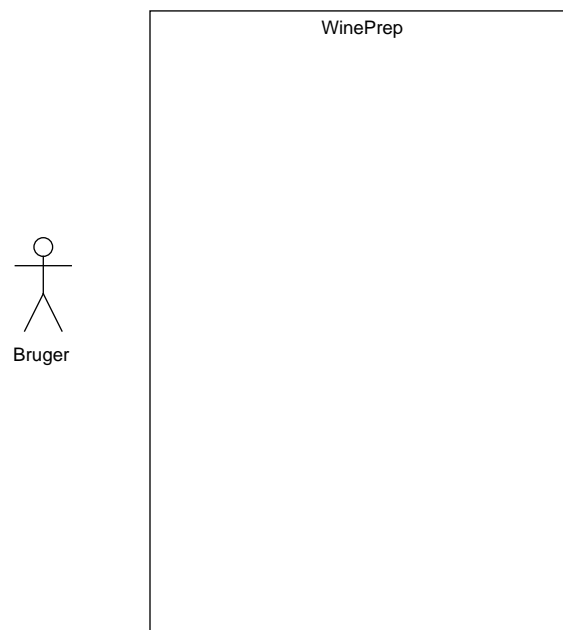
Indledning

1.1 projektformulering

Kapitel 2

Kravspekifikation

2.1 Aktør-Kontekst



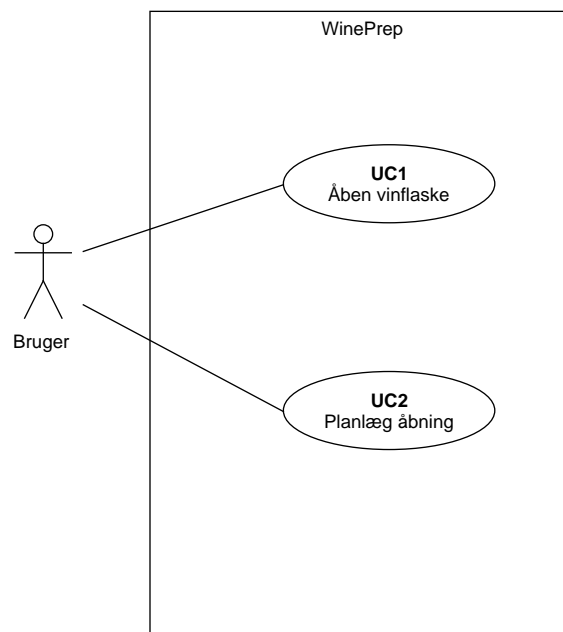
Figur 2.1: Aktør-kontekst Diagram

Aktør Beskrivelser

Bruger: Brugeren er systemets primære aktør. Brugeren er ham eller hende der betjener systemet, og har en opgave som ønskes løst af systemet.

2.2 Use-Cases

Use-case Diagram



Figur 2.2: Usecase Diagram

Use-case 1: Åbn Vin

Navn	UC 1: Åbn Vinflaske
Mål	At åbne vinflasken og dermed tillade brugeren adgang til vinen
Initiering	test 2
Aktører	Primær: Brug
Antal Samtidige forekomster	1
Prækondition	Vinflasken er anbragt i systemet og systemet er klar til brug. Desuden er vinflasken uåbnet og forseglingen er fjernet.
Postkondition	Vinflasken er åbnet
Hovedscenarie	<ol style="list-style-type: none"> 1. System detektere vinflaskens type og position [Ext. 1: System registrerer ugyldig type af vinflaske] [Ext. 2: System kan ikke registrere en vinflaske] 2. System låser vinflasken i dens position 3. System fjerner prop fra vinflasken 4. System frigiver vinflasken 5. System meddeler brugeren om at vinflasken er åbent og klar til brug. 6. System dispenserer prop.

Udvidelser/Undtag

Ext.1 System registrerer ugyldig type af vinflaske

[1.1] System meddeler brugeren om at typen af vinflaske er ugyldig.

[1.2] UC1 Afsluttes.

Ext.2 System kan ikke registrere en vinflaske

[2.1] System meddeler brugeren om at ingen vinflaske er registreret

[2.2] UC afsluttes

Use-case 2: Planlæg Åbning

Navn	UC 2: Planlæg Åbning
Mål	Vinen er drikkeklar til et forudbestemt tidspunkt
Initiering	Bruger trykker <i>Planlæg Åbning</i> på brugergrænsefladen
Aktører	Primær: Bruger
Antal Samtidige forekomster	1
Prækondition	Vinflasken er anbragt i systemet og systemet er klar til brug. Desuden er vinflasken uåbnet og forseglingen er fjernet.
Postkondition	Vinflasken er drikkeklar til det valgte tidspunkt
Hovedscenarie	<ol style="list-style-type: none"> 1. Bruger vælger tidspunkt på systemet [Ext. 1: Bruger ønsker ikke at åbne vin] 2. Bruger bekræfter valgt tidspunkt [Ext. 2: Vinen kan ikke iltes korrekt til det valgte tidspunkt] 3. System venter til iltningstidspunktet [Ext. 3: System registrerer ugyldig type af vinflaske] 4. System detekterer vinflaskens type og position [Ext. 4: System registrerer ugyldig type af vinflaske] [Ext. 5: System kan ikke registrere en vinflaske] 5. System låser vinflasken i dens position 6. System fjerner prop fra vinflasken 7. System frigiver vinflasken 8. System dispenserer prop 9. System venter til, at vinen er drikkeklar 10. System meddeler brugeren om, at vinen er drikkeklar

Udvidelser/Undtag

Ext.1 Bruger ønsker ikke at åbne vin

[1.1a] Bruger trykker på *Tilbage*

[1.2b] UC afsluttes

Ext.2 Vinen kan ikke iltes korrekt til det valgte tidspunkt

[2.1] System beder bruger bekræfte valg af tidspunkt

[2.2a] Bruger trykker bekræft

[2.3a] UC fortsættes fra punkt 1 i UC 1

[2.2b] bruger trykker *Annuller*

[2.3b] UC afsluttes

Ext.3 Bruger annullerer planlagt åbning af vin

[3.1] Bruger trykker *STOP!*

2.3 Ikke-funktionelle krav

Brugervenlighed

1. De virtuelle knapper på systemets grafiske brugergrænseflade skal have et areal på min 2.5 x 2.5 cm.
2. Systemet skal give brugeren besked om vinens status via tekst på touch skærmen.

Ydelse

1. Når systemet tændes, skal det kunne starte op, og være klar til modtage brugerinput på max. 2 minutter.
2. Den grafiske brugergrænsefladen skal have en reaktionstid på max 1 sek fra brugerinput via touchskærmen til opdatering af det grafiske layout.
3. Systemet skal kunne starte motorer til fastlåsning af vinen indenfor max 5 sekunder efter brugerinput "Åben nu" på brugergrænsefladen, og kunne færdiggøre åbningen af vinen indenfor max 1 minut efter brugerinput.
4. Når brugeren vælger "Planlæg åbning", skal systemet kunne åbne vinflaksen med en afvigelse på max 1 minut fra det indstillede åbningstidspunkt. Her skal åbning af vinen ligeledes kunne færdiggøres af systemet på max 1 minut.

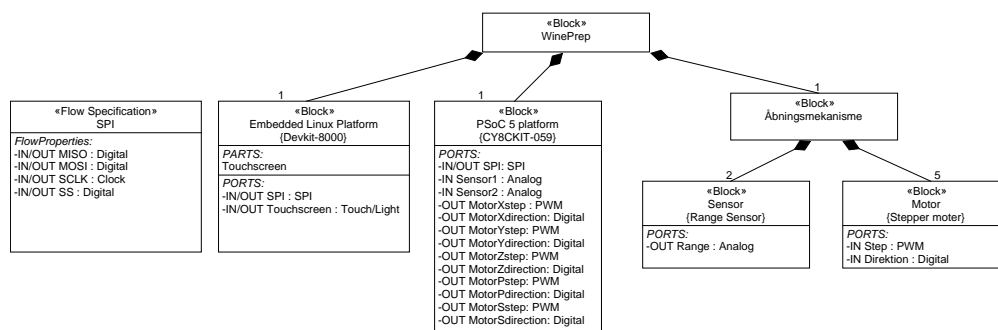
Vedligeholdelse

1. Koden til systemet skal skrives i programmerings sprogene c og c++.
2. Systemet skal betjenes via et embedded system hvorpå en Linux platform er installeret.
3. Motor- og sensorstyring skal foregå via en PSoC.

Kapitel 3

Hardware Arkitektur

Block Definition Diagram



Figur 3.1: BDD for WinePrep

Blok beskrivelser

Her følger beskrivelser af de enkelte blokke på vores BDD, se side 7 Figur 3.1.

WinePrep blokken er det samlede system der består af underblokkende Embedded Linux Platform, PSoC 5 Platform samt Åbningsmekanismen.

Embedded Linux Platform Dette er den blok der håndtere brugerens interaktion med systemet. Blokken består af et Devkit800 med touchskærm. Som styresystem på platformen anvendes der Linux distributionen Ångström. Her fra anvendes der QT til at lave den grafiske brugerflade der vises på touchskærmen til brugeren af systemet. Samtidig kommunikerer Embedded Linux Platformen med vores PSoC 5 Platform via SPI standarden.

PSoC 5 Platform PSoC 5 baseret platform der står for styring af Motor og Sensor blokkene, samt kommunikere med blokken Embedded Linux Platform over SPI.

Motor Denne blok definere de aktuatorer der anvendes til at trække proppen op af vinen samt til at sikre at åbningsmekanismen rammer vinflasken korrekt uden at vælge eller beskadige vinen.

Sensor Afstandssensorer til detektering af vinflaskens placering samt størrelse, så åbningsmekanismen ud fra dette kan positioneres korrekt ved hjælp af motorer på X, Y , Z akserne.

Ting der først bliver fast besluttet på senere iterationer/sprints

Motor Valg er ikke 100% fastlagt, hvorfor det her i BDD modelleres med step-per motors, og portene er derfor heller ikke 100% korrekte da dette afhænger af motorstyringen.

Sensor typer og antal ligger kun delvist fast, der vil være minimum 2 afstandssensorer af en art, hvorfor BDD er lavet mod 2 sensorer, det bliver enten lys baserede eller lydbaserede, der vil give et analog output signal i form af en spænding der afhænger af afstanden.

Kapitel 4

Software Arkitektur