

# Spesifikasjoner

## Om dokumentet

### *Mål med dokumentet*

*Dette dokumentet er ment for å formidle hvordan applikasjonen skal oppføre seg, hvilke funksjonaliteter som må implementeres for å oppnå dette, og hvordan disse funksjonalitetene blir implementert.*

*Kravene forbedres og refaktoreres etter hvert som applikasjonen vokser. Ikke tenk på dette dokumentet som et "ferdig" dokument.*

### *Dokumentets mulige forbedringer*

- Splitt opp kravene for å unngå flere brukerhistorier unødvendig på samme krav.

## Innhold

<b>Om dokumentet .....</b>	<b>1</b>
<b>Mål med dokumentet .....</b>	<b>1</b>
<b>Dokumentets mulige forbedringer .....</b>	<b>1</b>
<b>Klassebeskrivelser: .....</b>	<b>2</b>
<b>Om definisjon av krav.....</b>	<b>3</b>
Formatet krav skrives på: .....	3
<b>Spillbrett .....</b>	<b>3</b>
<b>KRAV: Vise et spillbrett .....</b>	<b>3</b>
Brukerhistorie:.....	3
Akseptansekriterier: .....	3
Arbeidsoppgaver: .....	3
<b>KRAV: Vise brikke på spillbrett.....</b>	<b>3</b>
Brukerhistorie:.....	3
Akseptansekriterier: .....	4
Arbeidsoppgaver: .....	4
<b>KRAV: Flytte brikke .....</b>	<b>4</b>
Brukerhistorie:.....	4
Akseptansekriterier: .....	4
Arbeidsoppgaver: .....	4
<b>KRAV: Robot besøker flagg .....</b>	<b>4</b>
Brukerhistorie:.....	4
Akseptansekriterier: .....	4
Arbeidsoppgaver: .....	4

<b>KRAV: Robot vinner ved å besøke flagg</b> .....	<b>5</b>
Brukerhistorie:.....	5
Akseptansekriterier: .....	5
Arbeidsoppgaver: .....	5
<b>Kort</b> .....	<b>5</b>
<b>KRAV: Dele ut kort</b> .....	<b>5</b>
Brukerhistorie:.....	5
Akseptansekriterier: .....	5
Arbeidsoppgaver: .....	5
<b>KRAV: Velge 5 kort</b> .....	<b>6</b>
Brukerhistorie:.....	6
Akseptansekriterier: .....	6
Arbeidsoppgaver: .....	6
<b>KRAV: Bevege robot ut fra valgte kort</b> .....	<b>6</b>
Brukerhistorie:.....	6
Akseptansekriterier: .....	6
Arbeidsoppgaver: .....	6
<b>GUI</b> .....	<b>6</b>
<b>KRAV: GUI til meny</b> .....	<b>6</b>
Brukerhistorie:.....	7
Akseptansekriterier: .....	7
Arbeidsoppgaver: .....	7
<b>KRAV: GUI til game som håndterer kort</b> .....	<b>7</b>
Brukerhistorie:.....	7
Akseptansekriterier: .....	7
Arbeidsoppgaver: .....	7
<b>KRAV: Options meny for å skifte innstillinger</b> .....	<b>7</b>
Brukerhistorie:.....	7
Akseptansekriterier: .....	7
Arbeidsoppgaver: .....	7
<b>Multiplayer</b> .....	<b>8</b>
<b>KRAV: Spille fra flere maskiner</b> .....	<b>8</b>
Brukerhistorie:.....	8
Akseptansekriterier: .....	8
Arbeidsoppgaver: .....	8
<b>KRAV: Hoste et spill</b> .....	<b>8</b>
Brukerhistorie:.....	8
Akseptansekriterier: .....	8
Arbeidsoppgaver: .....	8
<b>KRAV: Joine et spill</b> .....	<b>8</b>
Brukerhistorie:.....	8
Akseptansekriterier: .....	8
Arbeidsoppgaver: .....	9

Klassebeskrivelser:

*I kravene refererer vi til klasser, her skriver vi informasjon om de mest sentrale klassene.*

- GameRunner: Koordinerer input og backend med frontend.
- GameScreen: Tegner in-game GUI. (Brett deriblant)
- RRApplication: Baseklassen for spillet. Fundamentet.
- Map: Kommuniserer valgt brett og viktig brett-informasjon.
- Grid: Den logiske representasjonen av spillbrettet og elementer der.

## Om definisjon av krav

*Kravene til applikasjonen.*

Formatet krav skrives på:

1. Brukerhistorie: Ønsket oppførsel, grunnlag for å ha krav i programmet og hvem som har bruk for det.
2. Akseptansekriterier: Hvilke delegenskaper programmet må ha for å oppfylle brukerhistorien.
3. Arbeidsoppgaver: Klasser og metoder som brukes for å implementere de ønskede egenskapene til kravet. Bryter ned krav til deloppgaver.

## Spillbrett

*KRAV: Vise et spillbrett*

**Brukerhistorie:**

Som en spiller ønsker jeg å se spillbrettet for å orientere meg.

**Akseptansekriterier:**

- Når man velger spill fra menyen, skal spillbrettet vises på skjermen.

**Arbeidsoppgaver:**

*Klasser som brukes for å oppfylle akseptansekriteriene er:*

- GameRunner, GameScreen, RRApplication, Map.

*Med dem så må man:*

- Hente informasjon om hvilket kart man bruker.
- Hente informasjon om elementene på kartet (brikker, annet).
- Tegne dette på kartet.

*Sentrale metoder er:*

```
public String getMapPath() // I map.
```

*KRAV: Vise brikke på spillbrett*

**Brukerhistorie:**

Som en spiller ønsker jeg å se egen brikke på bretter for å vite hvor jeg er.

**Akseptansekriterier:**

- Alle skal kunne se brikken til spilleren på spillebrettet.

**Arbeidsoppgaver:**

*Klasser som brukes for å oppfylle akseptansekriteriene er:*

- GameRunner, GameScreen, RRAplication, Map.

*Med dem så må man:*

- Hente spillerposisjon (GameRunner).
- Tegne spiller på kartet (GameScreen).

## *KRAV: Flytte brikke*

**Brukerhistorie:**

Som en utvikler ønsker jeg å kunne bevege spillebrikken i spillet med piltaster for å teste brett funksjoner lettere

**Akseptansekriterier:**

- Når en piltast trykkes bevegtes spillebrikken i en retning.
- Hvis spilleren trykker høyre, vil robotens x-verdi øke med 1.
- Hvis spilleren trykker venstre, vil robotens x-verdi synke med 1.
- Hvis spilleren trykker opp, vil robotens y-verdi øke med 1.
- Hvis spilleren trykker ned, vil robotens y-verdi synke med 1.

**Arbeidsoppgaver:**

*Klasser som brukes for å oppfylle akseptansekriteriene er:*

- GameRunner: Koordinerer input og backend med frontend.
- GameScreen: Tegner in-game GUI. (Brett deriblant)
- RRAplication: Baseklassen for spillet. Fundamentet.
- Game: Holder Grid, informasjon om spillet og annen funksjonalitet.

*Med dem så må man:*

- Ta input og kommunisere til backend. (GameScreen og GameRunner).
- Oppdatere posisjoner (Game -> Grid).
- Tegne spiller på kartet (GameScreen).

## *KRAV: Robot besøker flagg*

**Brukerhistorie:**

Som robot vil jeg besøke flagg for å få poeng.

**Akseptansekriterier:**

- Gitt at en robot passere et flaggs koordinater, skal roboten kåres som vinner om flagget er det siste i sekvensen av flagg man må passere for å vinne.
- Gitt at robot lever og passerer et flagg (i riktig rekkefølge), skal en flagg-counter økes.

**Arbeidsoppgaver:**

*Klasser som brukes for å oppfylle akseptansekriteriene er:*

- Robot, Flag, Game, Grid.

*Med dem så må man:*

- Registrere om en robot passerer et flagg om flagget er neste i rekkefølgen.
- Sjekke om robot har vunnet (passert alle flaggene).

*Sentrale metoder er:*

*KRAV: Robot vinner ved å besøke flagg*

**Brukerhistorie:**

Som robot vil jeg vinne ved å besøke alle flaggene i riktig rekkefølge for at spillet skal ha et mål.

**Akseptansekriterier:**

- Gitt at en robot har passert flaggene i riktig rekkefølge ( $f_1 \dots f_{n-1}$ ) skal roboten vinne når den når flagg  $f_n$ .
- Roboten har en liste over flagg den har besøkt i riktig rekkefølge

**Arbeidsoppgaver:**

*Klasser som brukes for å oppfylle akseptansekriteriene er:*

- Flag, Robot, Game, Grid.

*Med dem så må man:*

- Sjekke om robot har vunnet.

*Sentrale metoder er:*

`checkIfLocalHasWonOrLost()`

## Kort

*KRAV: Dele ut kort*

**Brukerhistorie:**

Som et spill vil jeg at spillerne skal få utdelt tilfeldige kort for å bevege robotene sine.

Som et spill vil jeg at kortstokken tømmes etter hvert som spillerne får kort for at spillerne ikke får alle de samme kortene. For å kunne dele ut kort må spillet først ha en kortstokk og minst en spiller som kan motta kort. Dette skjer automatisk som rundene går.

**Akseptansekriterier:**

- Gitt at kort har blitt delt ut, skal en spiller ha en liste med mottatt kort.

**Arbeidsoppgaver:**

*Klasser som brukes for å oppfylle akseptansekriteriene er:*

- CardDealer
- CardDeck
- Player

*Med dem så må man:*

- Instansiere en ny deck-list i Player, og kalle CardDealer sin deal metode.

*Sentrale metoder er:*

### *KRAV: Velge 5 kort*

#### **Brukerhistorie:**

Som en spiller vil jeg kunne velge hvilke 5 kort som skal spilles i ønskelig rekkefølge.

Som et spill vil jeg ikke at spillerne skal ha mer enn 5 kort slik at en spiller ikke kan ha alle kortene samtidig.

#### **Akseptansekriterier:**

- Gitt at spilleren velger et kort, skal det bli lagt til i sin egen hånd-liste.
- Gitt at spilleren holder 5 kort, kan han ikke velge flere.
- Gitt at spilleren velger kort, skal kortet fra "9-kort" listen forsvinne, så man ikke kan ta duplikater.

#### **Arbeidsoppgaver:**

*Klasser som brukes for å oppfylle akseptansekriteriene er:*

- CardDealer, Player

*Med dem så må man:*

- Kalle på CardDealer's dealCards
- Player må ha en liste for valgte og mulige kort.

*Sentrale metoder er:*

### *KRAV: Bevege robot ut fra valgte kort*

#### **Brukerhistorie:**

Som en spiller ønsker jeg å gi instruksjoner til roboten med kortene mine, for at den skal bevege seg.

#### **Akseptansekriterier:**

- Roboten flyttes etter kortets instruksjoner

#### **Arbeidsoppgaver:**

*Klasser som brukes for å oppfylle akseptansekriteriene er:*

- Robot, Grid, Card-package.

*Med dem så må man:*

- Robot sin posisjon eller retning i Grid må endres ved hjelp av metoden i TurnCard og StepCard.

*Sentrale metoder er:*

## GUI

### *KRAV: GUI til meny*

#### Brukerhistorie:

Som spiller vil jeg kunne se et grafisk brukergrensesnitt, der jeg kan navigere ved hjelp av piltaster eller peker som gjør opplevelsen mer brukervennlig.

#### Akseptansekriterier:

- Gitt at man velger på en av knappene, skal input bli registrert og du skal få tydelig feedback på knappen fungerer.

#### Arbeidsoppgaver:

*Klasser som brukes for å oppfylle akseptansekriteriene er:*

- Applistener, MainMenuScreen

*Med dem så må man:*

*Sentrale metoder er:*

### *KRAV: GUI til game som håndterer kort*

#### Brukerhistorie:

Som spiller ønsker jeg å kunne se kortene jeg har blitt tildelt og ha tilgang til å velge kort.

#### Akseptansekriterier:

- Gitt at jeg velger kort i en viss rekkefølge, skal funksjonene bli eksekvert i riktig rekkefølge.
- Det skal ikke gå an å velge samme kort flere ganger.

#### Arbeidsoppgaver:

*Klasser som brukes for å oppfylle akseptansekriteriene er:*

*Med dem så må man:*

*Sentrale metoder er:*

### *KRAV: Options meny for å skifte innstillinger*

#### Brukerhistorie:

Som spiller vil jeg ha tilgang til innstillinger som å skifte lyd og display mode.

#### Akseptansekriterier:

- Fullskjerm-knapp skifter bildet.
- Mute demper musikk.

#### Arbeidsoppgaver:

*Klasser som brukes for å oppfylle akseptansekriteriene er:*

- OptionsScreen

*Med dem så må man:*

*Sentrale metoder er:*

# Multiplayer

## *KRAV: Spille fra flere maskiner*

### Brukerhistorie:

Som spill-server vil jeg ha oversikt over alle spillers kort og roboter, og kunne sende informasjon til klienter om hvordan brettet ser ut og spillere befinner seg.

### Akseptansekriterier:

- Gitt av en spiller beveger seg på sin skjerm, skal det samme skje på andre brukers skjerm.
- Gitt at en spiller vinner vil alle få beskjed.

### Arbeidsoppgaver:

*Klasser som brukes for å oppfylle akseptansekriteriene er:*

- Client
- Server
- Game

*Med dem så må man:*

- Få alt i spillet til å snakke sammen!

*Sentrale metoder er:*

## *KRAV: Hoste et spill*

### Brukerhistorie:

Som en spiller ønsker jeg å kunne sette opp et spill hvor andre kan bli med.

### Akseptansekriterier:

- Når jeg hoste et spill startes, og andre skal ha tilgang til å koble seg til.

### Arbeidsoppgaver:

*Klasser som brukes for å oppfylle akseptansekriteriene er:*

- Server

*Med dem så må man:*

- Starte server.
- Sende spill-informasjon.

*Sentrale metoder er:*

## *KRAV: Joine et spill*

### Brukerhistorie:

Som spiller vil jeg kunne trykke Join i Multiplayer menyen for å bli med i et spill som en annen person hoster.

### Akseptansekriterier:



- Multiplayer meny må ha funksjoner til å kobles til en server.

Arbeidsoppgaver:

*Klasser som brukes for å oppfylle akseptansekriteriene er:*

- Client

*Med dem så må man:*

- Koble til ekstern server (en annen spiller).

*Sentrale metoder er:*