Innhold

[Mine sirkler – Ny android versjon 1](#_Toc338926129)

[Visjon 1](#_Toc338926130)

[Kravspec: 1](#_Toc338926131)

[Scope 2](#_Toc338926132)

[Teknikker som brukes 2](#_Toc338926133)

[Eksempel på brukergrensesnitt 3](#_Toc338926134)

[Design 4](#_Toc338926135)

[Klassediagram 4](#_Toc338926136)

[Eksempel på bruk av klassene 4](#_Toc338926137)

[Flytskjema 5](#_Toc338926138)

# Mine sirkler – Ny android versjon

## Kravspesifikasjon

### Visjon

Lage mineSirkler-app for Android som ser bra ut samtidig som den er enkel og praktisk

### Kravspec:

|  |  |  |
| --- | --- | --- |
| Pri. | Funksjonalitet | Niv |
| 1 | 6 Hovedkategorier med Subkategorier og Underemner til Subkategoriene. Underemnene skal være mulig å redigere |  |
| 1 | Navigering mellom menyer, knapper for frem og tilbake |  |
| 1 | Logge seg inn som administrator, for å få tilgang til redigering av kategorier. |  |
| 2 | Brukergrensesnitt for Hovedkategorier som ser ut som en sirkel |  |
| 2 | Grensesnittet skal være dynamisk. Sirklene skal være farget grå, helt til de er fylt inn med informasjon. Da blir de farget med gjeldende farge for kategorien. |  |
| 2 | Mulighet for å lage egendefinerte Hovedategorier og Underemner |  |
| 3 | Mulighet for å klippe ut en tekst fra et annet sted, og lime det inn i kategori-visningen(Tilgang til utklippstavle) |  |
| 3 | Hjelp funksjon med eksempeltekst for hver at kategoriene? |  |
| 2 | Mulighet å vise oversikten på to måter. En visning som sirkler, og en som liste |  |
| 2 | Mulighet for å se innholdet både vertikalt og horisontalt på oversiktlig måte |  |
| 3 | Settings meny der en kan velge hvordan innholdet skal vises |  |
| 2 | Mulighet for å eksportere informasjon som en har lagt inn til andre enheter |  |

### Scope

Oppdragsgiver for prosjektet er Ellen Kleven, som skal benytte produktet til å hjelpe Aspergere som trenger hjelp til å organisere livet sitt. Det skal legges ut som en gratisapplikasjon på Android-marketplace(Eventuellt betalingsversjon for å dekke utgifter). Brukere er aspergere som trenger et hjelpemiddel, eller støttepersoner som skal tilrettelegge for brukere. Det vil også bli tatt ibruk av Ellen Kleven når hun reiser rundt og holder foredrag.

### Teknikker som brukes

Jeg ønsker å lage programmet så enkelt så mulig, men samtidig med muligheter for å utvide det, og å kjøre på forskjellige platformer. I utgangspunktet vil jeg programmere det for android, men jeg vil også undersøke mulighetene for å utvikle det i Apache Flex, som er en cross-platform sdk som skal være enkel å lage programmer i. Cross-plaform, betyr at programmet skal kunne kjøres på alle operativsystem, Apple sitt, Microsoft Windows, og Android smarttelefoner og nettbrett.

**Android SDK Tools:** Verktøy som muliggjør utvikling for mobile applikasjoner. Inkluderer plattformen til Android, og en virituell mobil enhet for kjøring/debugging av programmet

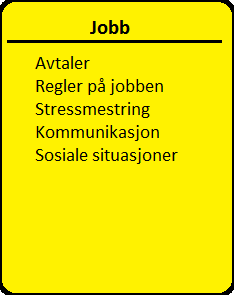
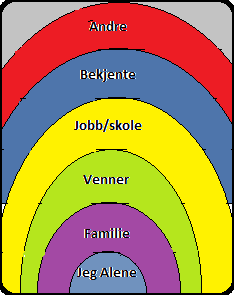
**Apache Flex:** Fordelen med Flex, er at det kan kjøres som en internett-applikasjon, eller en mobil-applikasjon, og at det kan kjøres på forskjellige platformer. Det er også implementet i java, så det kan programmeres i Eclipse Ulempen er at jeg ikke har noe kjennskap til utvikling med disse verktøyene. Det er i skrivende stund, kun 500 000 mennesker som programmerer i flex, og det er ikke mange opplæringsverktøy på internett som forklarer hvordan en bruker det sammen med Eclipse. Programmer som Flex er avhengig av er Adobe AIR integration kit for bygging og debugging av mobile applikasjoner, Adobe PixelBender Toolkit trengs også for å bygge noen filer som støtter effekter.

**Eclipse:** Utviklingsverktøy for java, med utvidelse for andre språk. Fordelen med eclipse, er at jeg har meget god kjennskap til bruk av verktøyet. Det har gode muligheter for debugging. Ulemper er at jeg ikke vet hvordan det fungerer med Android-utviklings plugins, og om disse fungerer bra sammen med Eclipse

**Java:** Fordelen med java er at det har verktøy for brukergrensesnitt. Og det brukes av flex og android.

**Word:** Word vil bli brukt for å lage dokumentasjon, spesifikasjoner og design. Dette i mangel av andre designverktøy, og fordi modellene vil bli ganske enkle. Sekvensdiagram og klassediagram-template for word:Grupperte objekter, klasser, og funksjonspiler m/tekst

### Eksempel på brukergrensesnitt



Det gamle brukergrensesnittet viser sirklene som en liste, men det er ønskelig å vise dem som en sirkel som ovenfor i stedet. Dette bør ikke gå utover brukervennligheten, så en bør bruke en løsning som gjør det enkelt og oversiktlig å navigere.

Et forslag til navigering er å ha to visninger av sirkelen. En der en viser sirkelen som ovenfor, med nedre skjerm grense «Jeg alene» sirkelen, og øverst «andre sirkelen». Brukeren bør her kun ha mulighet for å scrolle oppover dersom han har lagt til sirkler utenfor de 6 obligatoriske sirklene. Når brukeren klikker på en sirkel, vil han få opp det aktuelle dataarket med informasjon som vist ovenfor.

Den andre visningen er å vise hele sirkelen. Ulempene her er at brukerne kan trykke på feil sirkler da de blir for små. Den andre ulempen er dersom brukeren ønsker å legge til sine egne sirkler. Disse vil da komme utenfor de andre sirklene, og en vil bli nødt til å scrolle rundt på skjermen. Det er også et problem å implementere denne funksjonaliteten, dersom en legger til flere sirkler, og sletter en sirkel etterpå. Skal de ytre sirklene da, bevege seg ned et nivå og overskrive nivået til sirkelen som ble slettet, eller skal det bli et tomt nivå? Hva hvis brukeren vil flytte sine sirkler opp eller ned i nivåer?

Problemet med sirklene ovenfor er helt klart den vanskeligste utfordringen med programmet. Dersom en skal ha egendefinerte sirkler vil det bli veldig komplekst å lage grensesnittet. En løsning kan være å lage sirkelen med bare de 6 grunnsirklene, og heller ha en dropdown, meny der brukeren kan legge inn sine egne ‘sirkler’. Det kan være lurt å begrense antallet med egendefinerte sirkler, en bruker bør ikke ha bruk for så mange av disse mener jeg.

## Design

Lagring: Vi vil helst lagre informasjonen en lesbar tekstfil, slik at en kan lese informasjonen uten MineSirkler, og ha mulighet for å eksportere det til for eksempel word, excel eller andre enheter som har MineSirkler.

Det går også an å lagre på database. Fordelen med daabase, er at progammet blir raskere å utvikle og utvide siden en slipper å lage egne funksjoner for lagring sletting og redigeing. Det er enkelt å eksportere informasjonen fra databasen til en tekstfil også. Vi vil vurdere å ha en knapp for å eksportere informasjonen.

Arkitekturen vil være tredelt, 1. Grensesnitt, 2. Logisk Lag, 3 Database.

Grensesnittet består av alle skjemaene som brukeren har tilgang til. Dette er foreløpig MineSirkler og visningen for de underliggende dataarkene, men vil med tiden også inkludere grensesnitt for personlige innstillinger og redigering av data.

Logisk lag er resten av klassene som utgjør lagringsstrukturen for dataarkene. Det inkluderer også Controller-klassen som er bindeleddet mellom grensesnittet og resten av klassene. Controller-klassen sørger for at alle klassene blir opprettet riktig, og oppdatert når bruker skriver inn data.

Database-laget består av databasen og klassene som kommuniserer med databasen. Disse klassene er ansvarlig for å forsyne Controlleren med funksjoner som muliggjør tilgang mot databasen.

### Klassediagram

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  | | --- | | Controller | | id: int  navn2:string | | +LagInfokort(sirkel1,sirkel2, osv..)  +HentInfokort(infokortNr)  +SlettInfokort(infokortNr) | |  |  |  | | --- | | InfoKort | | infokortID: int  navn:string  beskrivelse:string  farge:string  hierarkiNivå:string | | +LeggTilUndernivå()  +SlettUndernivå() |      |  | | --- | | Interface | | meg:sirkelObjekt  familie:sirkelObjekt  venner:sirkelObjekt  osv… | | +MegClick(sirkelObjekt)  +FamilieClick(sirkelObjekt)  +VennerClick(sirkelObjekt) | |  | |

### Eksempel på bruk av klassene

Klassene designes etter controller-prinsippet, de vil si at brukergrensesnittet/skjemaene, kommuniserer kun med en klasse. Denne klassen har ansvaret for å kommunisere med alle klassene i det Logiske laget

Public class Controller()

Initialiser()

{

Infokort jobbInfo = New Infokort(1,Jobb, «Her er det kollegaer»,»gul»,1);

Infokort jobbinfoNivå2 = New Infokort(1,Lunch, «Her er det folk som snakker»,»gul»,2);

jobbInfo.LeggTilUnderNivå(jobbinfoNivå2);

}

### Flytskjema

#### Rediger infokort

Infokort

Controller

Interface

initialiser()

LastInfokort

Lag\_Infokort()

Kategori\_click()

RedigerKort()

|  |
| --- |
| MineSirkler.xml |
|  |
|  |