

# Food App

---

## Theme Project GR[10]

**Rasmus Holm Laursen(201270843), Karsten Michaelsen(11007),  
Andreas Lauridsen(10571), Nikolaj Stenvang Hansen(201270978)**

**1/6/2015**

## Indholdsfortegnelse

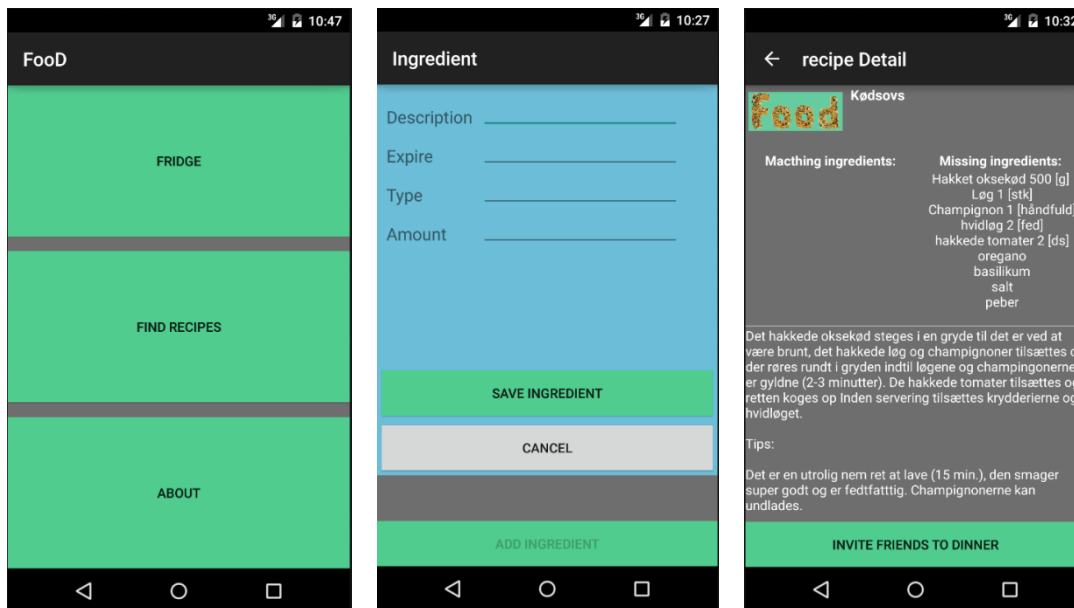
Indledning.....	2
Læringsmål der er opfyldt i opgaven.....	3
Design og funktionalitet af brugergrænsefladen. ....	4
Fridge Activity .....	4
Find Recipes Activity .....	5
Notifikation.....	6
Use Cases .....	7
Use Case diagram .....	7
Use case beskrivelser.....	7
Add ingredient.....	7
Edit ingredient .....	8
Delete ingredient.....	8
Inspect fridge.....	8
Suggest recipes .....	9
Invite friends to dinner .....	9
Klasse diagrammer .....	0
Sekvensdiagrammer .....	0
Ingredient activity.....	0
RecipeList Activity .....	1
Facebook activity .....	2
Konklusion .....	3
Arbejdsfordeling .....	3

## Indledning

Som det er beskrevet i synopsisen for projektet, er der arbejdet ud fra en vision som lyder følgende:

*"Visionen med dette projekt er at gøre den enkelte husstand mere bevidst om indholdet i køleskabet og anvendelsesmulighederne af disse til madlavning. På den måde kan man mindske madspild og sikre mere præcise indkøbsvaner."*

Denne vision har gennem hele projektet være omdrejningspunkt for udviklingen af Food App. Appens overordnet brugergrænse kan ses på Figur 1.



Figur 1 - Uddrag af forskellige dele af appens GUI

Appens funktionalitet tager udgangspunkt i brugerens køleskab, "Fridge". Det første der skal gøres når appen starter op, er at tilføje nye varer til ens virtuelle køleskab. Derved har appen mulighed for at forslå de opskrifter, som passer bedst i forhold til det brugeren har i sit køleskab, når der vælges "Find Recipes". I forbindelse med tilføjelse af vare, er det ligeledes muligt at slette og redigere allerede indtastede varer.

Når brugeren har fundet en passende opskrift vil vedkommende have mulighed for at invitere sine venner til at komme og spise med via et Facebook opslag. Der er desuden lavet en feature der viser en animation og afspiller en sang, når brugeren vælger at trykke på "About" knappen på forsiden. Denne del er dog ikke direkte sammenhængende med resten af projektet og er mest lavet som en lille sjov feature og er derfor ikke beskrevet dybdegående i denne rapport.

## Læringsmål der er opfyldt i opgaven

Til opgaven er der opstillet en lang række læringsmål, som der fra kurset side er forventning om appen indeholder så vidt det er muligt. I denne opgave er der opfyldt følgende læringsmål:

- Activity og Fragments
  - Kommunikation mellem aktiviteter og fragmenter. Eks. ved tilføjelse af en vare, her inflater aktiviteten "Ingredient" et fragment kaldet "Fragment\_Display\_Ingredient".
- Externalization
  - Dette er blandt andet benyttet i forbindelse med Styles, som er implementeret på eks. Knapper.
- Intents
  - I forbindelse med styring af aktiviteter er der benyttet intents.
- SQLite Database
  - Alt data til opskrifter og lagring af ingredienser er gemt på telefonens Internal storage via SQLite.
- Background tasking
  - En intent service bruges til at overvåge expire date og kommer med en notifikation, når noget er ved at blive for gammelt.
- Notification/Toasts
  - Benyttes til at notificere brugeren, når der registreres at en vare i køleskabet er ved at være udløbet.

## Design og funktionalitet af brugergrænsefladen.

Appen er designet med henblik på at være let tilgængelig og hurtig at lære. Derfor er der benyttet et design med få knapper på forsiden, se Figur 2. Her findes der tre knapper "Fridge", "Find recipes" og "About" som hver åbner en ny aktivitet, hvori deres funktionalitet er implementeret.

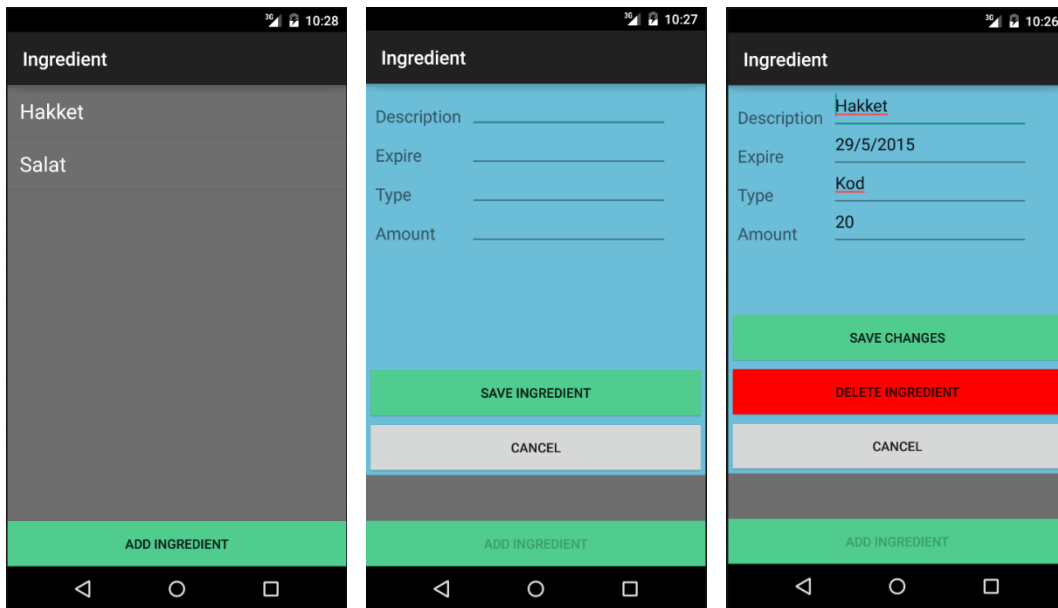


Figur 2 - Forside/Startside af Food App

### Fridge Activity

I aktiviteten Fridge får brugeren vist en liste af alle ingredienser som køleskabet indeholder. Hele gui'en kan ses i Figur 3. Her har brugeren mulighed for at tilføje nye varer via "Add ingredient", hvorefter fragmentet til tilføjelse af vare åbner. I dette fragment er der mulighed for enten at gemme de indtastede data via "Save ingredient" eller fortryde ved at trykke på "Cancel". Hvis det trykkes på en af varerne i listen vil der ligeledes åbnes et nyt fragment, hvor det er muligt enten at opdatere den valgte vare med nyt data ("Save changes") eller slette varen på "Delete ingredient". Det er også muligt at annullere via "Cancel". De data typer der er mulige at ændre på er.

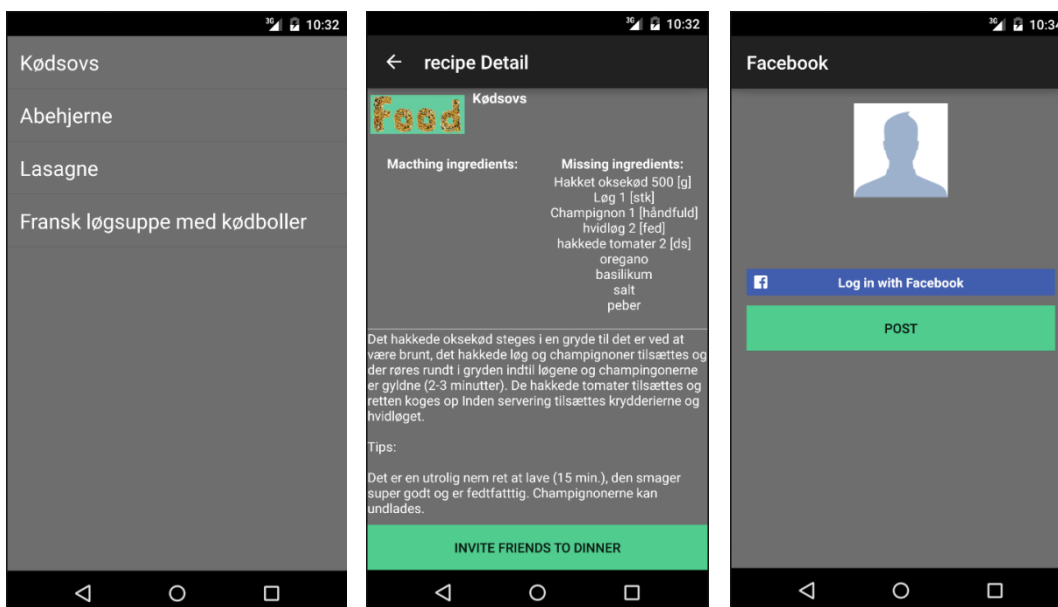
- **Description** - Hvilket er navnet på den ingrediens der ønskes at tilføje
- **Expire** - Hvor der åbnes en kalender til dato valg
- **Type** - Hvilken type ingrediens er der tale om. Eks. Mejeri eller grøntsager
- **Amount** - Hvor meget af den pågældende ingrediens har brugeren



Figur 3 - Gui for Fridge. List af vare - Tilføjelse af vare - Redigering af vare

## Find Recipes Activity

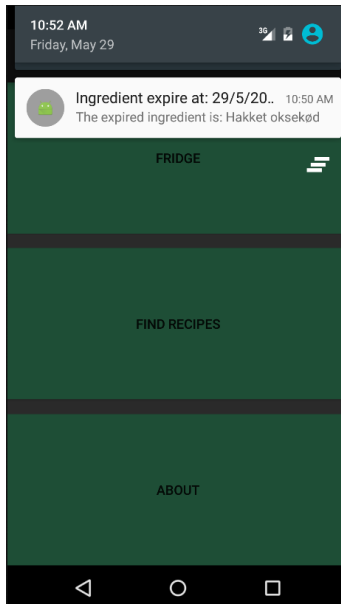
Når brugeren vælger at trykke på "Find Recipes" på forsiden åbnes en ny aktivitet. Denne indeholder en liste over alle de opskrifter som matcher ingredienser med opskrifter. Det kan eks. Være at brugeren har hakket oksekød i sit køleskab, hvorefter appen bl. a. vil foreslå kødsovs. På Figur 4 nedenfor kan gui'en ses. Hvis brugen trykker på en af de fundne opskrifter vil detaljerne for denne opskrift vises. Her kan der både ses hvilke ingredienser man allerede har og hvilke der mangler for at kunne lave retten. Inde i opskriften er det ligeledes muligt at invitere sine Facebook venner til at komme via knappen "Invite Friends To Dinner". Heri kan brugeren logge ind på Facebook og fortælle sine venner at de gerne må komme og spise eks. Kødsovs.



Figur 4 - Liste over fundne opskrifter - Detaljeret visning af opskrift - Invitere venner via Facebook

## Notifikation

Alle varer tjekkes dagligt om de er ved at udløbe. Hvis dette skulle være tilfældet vil nedenstående notifikation komme, se Figur 5. Den indeholder hvilken vare samt hvilken dato denne udløber.

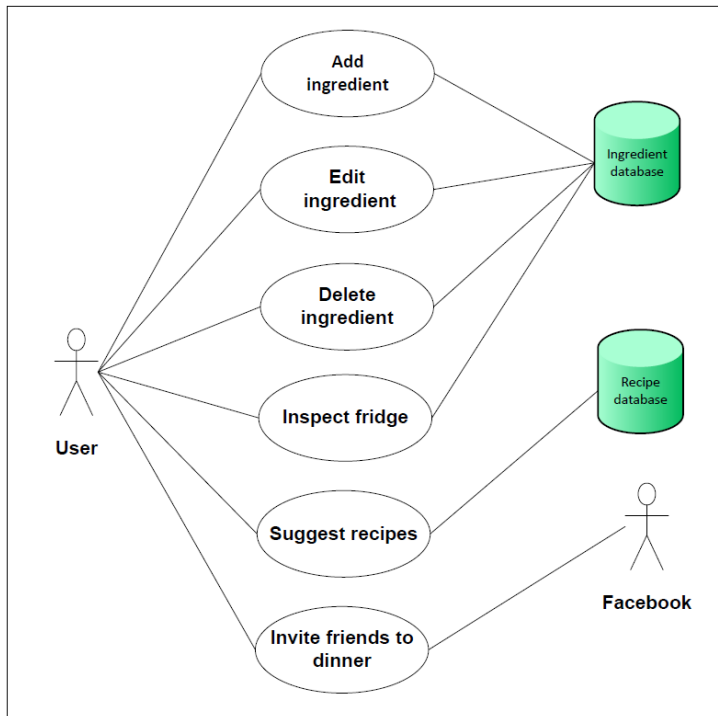


Figur 5 - Notifikation ved overskredet udløbsdato

## Use Cases

### Use Case diagram

Der er lavet et use case diagram over alle appens use cases, se Figur 6. Som aktør er der Brugeren (User), Facebook, Recipe database og Ingredient database.



Figur 6 - Use case diagram for Food App

Use casene er beskrevet i detaljer nedenfor.

### Use case beskrivelser

#### Add ingredient

Add Ingredient	
Goal	Brugeren har tilføjet en vare til køleskabet
Initiation	User
Precondition	Appen er åben
Postcondition	Varen er succesfuldt tilføjet til køleskabet
Main Scenario	<ol style="list-style-type: none"> <li>1. Brugeren vælger "fridge" fra hovedmenuen.</li> <li>2. Brugers aktuelle varer i køleskabet vises på en liste. Brugeren vælger "Add Ingredient".</li> <li>3. En ingredient dialog åbner, hvori brugeren angiver description, expiredate, type og amount af varen.</li> <li>4. Brugeren vælger "save ingredient".</li> <li>5. Databasen opdateres med den angivne vare.</li> </ol>
Extension	Extension 2a: Brugeren vender tilbage til main screen Extension 3a: Brugeren vender tilbage til køleskab



### Edit ingredient

Edit Ingredient	
Goal	Brugeren har redigeret en vare fra køleskabet
Initiation	User
Precondition	Appen er åben og en vare er tilføjet køleskabet
Postcondition	Varen er succesfuldt redigeret i køleskabet
Main Scenario	<ol style="list-style-type: none"> <li>1. Brugeren vælger "Fridge" fra hovedmenuen .</li> <li>2. Brugers aktuelle varer i køleskabet vises på en liste, hvorfra en vare vælges.</li> <li>3. En redigeringsdialog åbner hvor varens aktuelle tilstand vises. Brugeren redigerer i informationen omkring varen og vælger "save changes".</li> <li>4. Varen opdateres i databasen.</li> </ol>
Extension	Extension 2a: Brugeren vender tilbage til main screen Extension 3a: Brugeren vender tilbage til køleskab

### Delete ingredient

Delete Ingredient	
Goal	Brugeren har slettet en vare fra køleskabet
Initiation	User
Precondition	Appen er åben og en vare er tilføjet køleskabet
Postcondition	Varen er succesfuldt slettet fra køleskabet
Main Scenario	<ol style="list-style-type: none"> <li>1. Brugeren vælger "Fridge" fra hovedmenuen .</li> <li>2. Brugers aktuelle varer i køleskabet vises på en liste, hvorfra en vare vælges.</li> <li>3. En redigeringsdialog åbner hvor varens aktuelle tilstand vises. Brugeren vælger "delete ingredient".</li> <li>4. Varen slettes fra databasen.</li> </ol>
Extension	Extension 2a: Brugeren vender tilbage til main screen Extension 3a: Brugeren vender tilbage til køleskab

### Inspect fridge

Inspect Fridge	
Goal	Brugeren ser en liste af varer i køleskabet
Initiation	User
Precondition	Appen er åben og mindst én vare er tilføjet køleskabet
Postcondition	Brugeren ser en liste af alle vare i køleskabet
Main Scenario	<ol style="list-style-type: none"> <li>1. Brugeren vælger "Fridge" fra hovedmenuen.</li> <li>2. Brugeren får vist en liste af køleskabets indhold.</li> </ol>
Extension	Extension 2a: Brugeren vender tilbage til main screen

### Suggest recipes

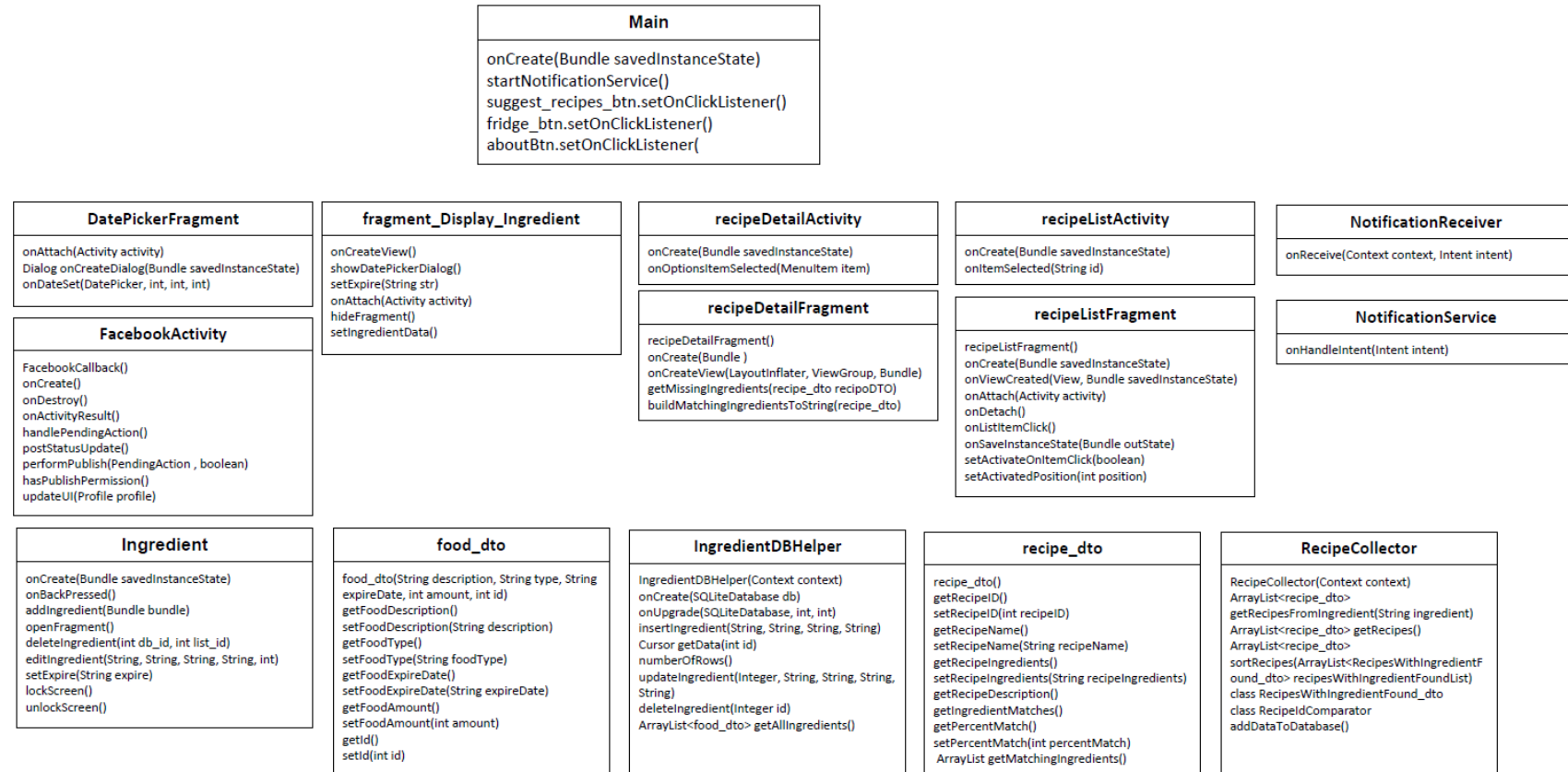
Suggest Recipes	
Goal	Brugeren får foreslået en liste af opskrifter ud fra indholdet i køleskabet
Initiation	User
Precondition	Appen er åben og mindst én vare er tilføjet til køleskabet
Postcondition	Brugeren har fået en liste af opskrifter, baseret på indholdet i køleskabet
Main Scenario	<ol style="list-style-type: none"> <li>1. Brugeren vælger "Find Recipes" fra hovedmenuen.</li> <li>2. Brugeren får vist en liste af opskrifter ud fra indholdet i køleskabet. Listen er rangeret efter flest matchene ingredienser fra køleskabet målt i procent.</li> <li>3. Brugeren vælger en opskrift fra listen.</li> <li>4. En detaljeret opskriftsdialog åbner med information om den valgte opskrift.</li> </ol>
Extension	Extension 2a: Brugeren vender tilbage til main screen

### Invite friends to dinner

Invite Friends To Dinner	
Goal	Brugeren har inviteret venner til mad via facebook
Initiation	User
Precondition	Appen er åben og en opskrift er valgt via use case "Suggest Recipes"
Postcondition	Venner bliver tagget i et facebookopslag med en invitation til aftensmad
Main Scenario	<ol style="list-style-type: none"> <li>1. Brugeren vælger "Invite Friends To Dinner" fra den detaljerede opskriftsdialog.</li> <li>2. Brugeren skriver et opslag omkring aftensmaden og tagger de venner der ønskes inviteret.</li> <li>3. Brugeren poster invitationen.</li> </ol>
Extension	Extension 2a: Brugeren vender tilbage til main screen Extension 2b: Brugeren skal logge ind først

## Klasse diagrammer

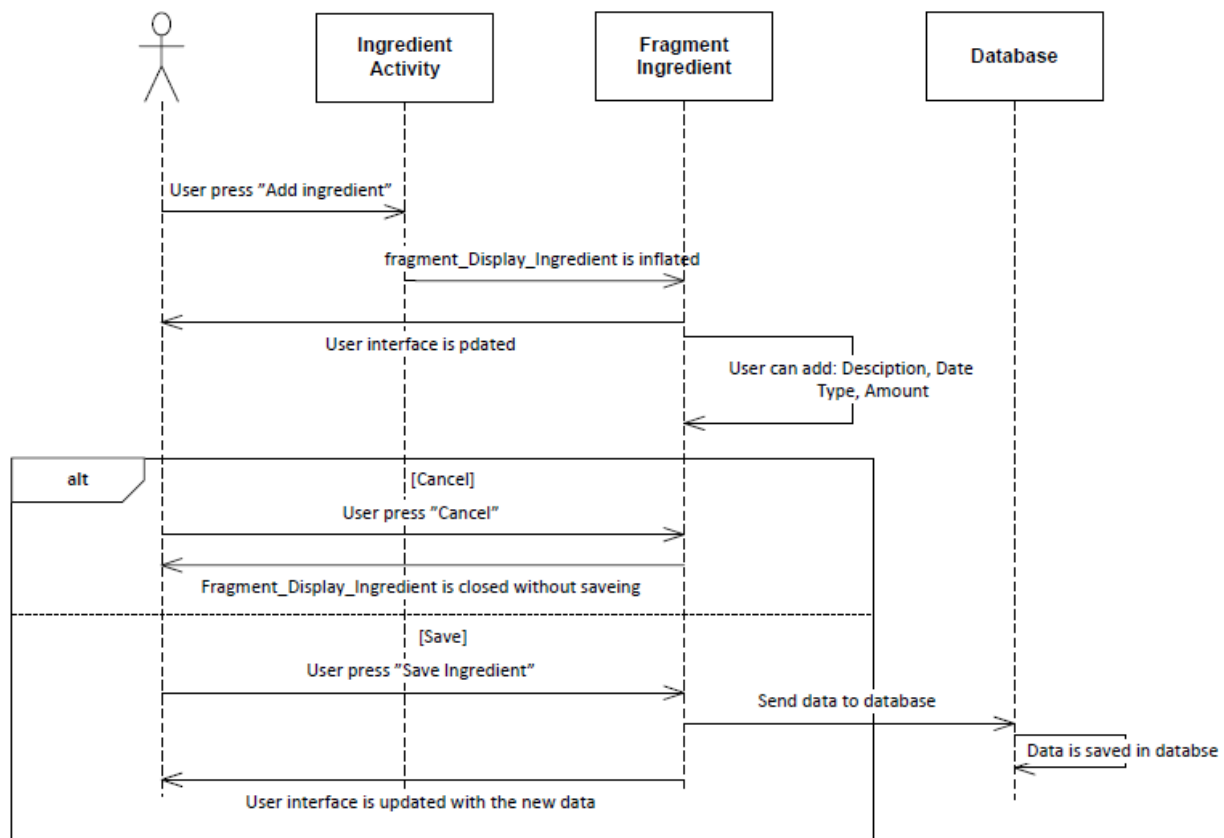
På nedestående figur 7 ses appens overordnede klassediagram, hvor de enkelte klassers metoder fremgår.



Figur 7 - Overordnet klasse diagram

## Sekvensdiagrammer

### Ingredient activity



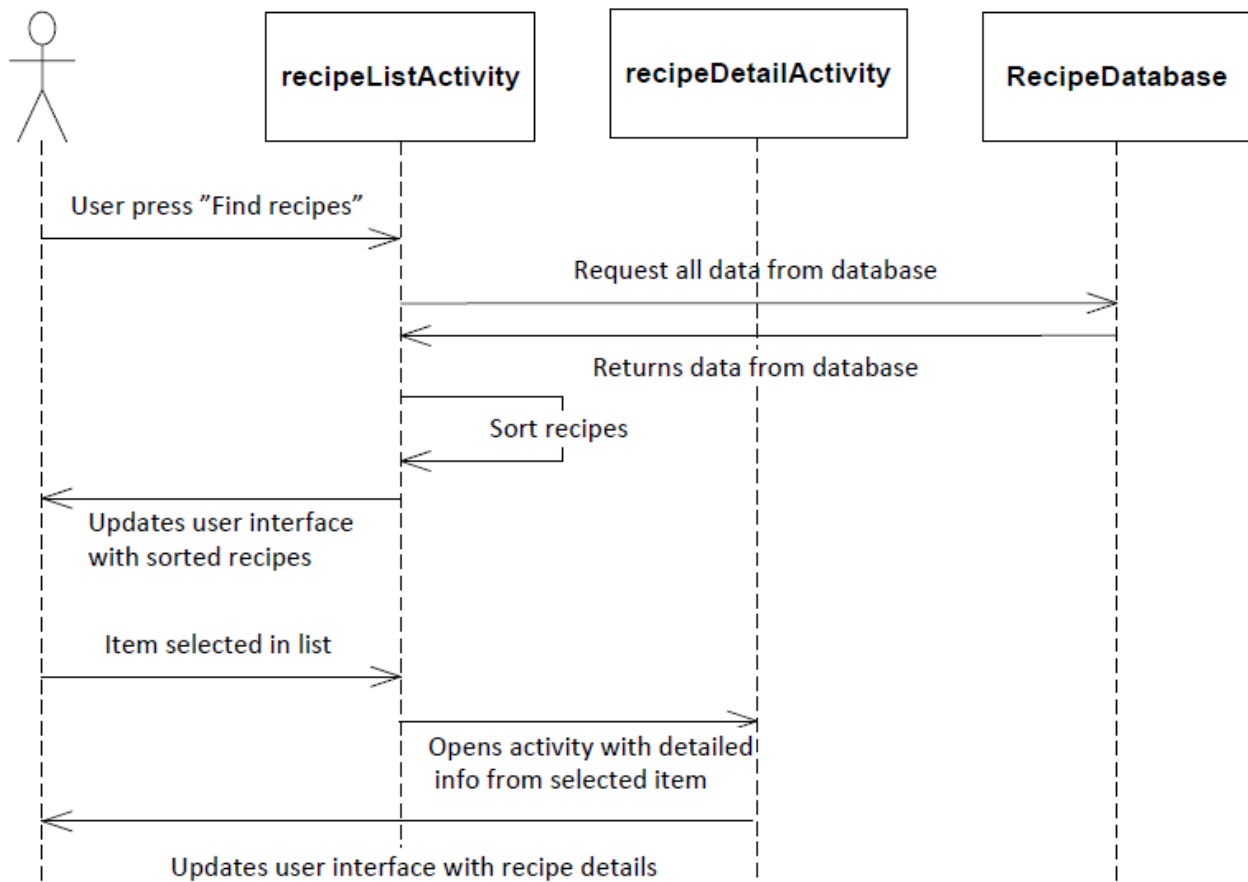
Figur 8 - Sekvensdiagram over Ingredient activity

Når brugeren vælger "Add Ingredient" vil Ingredient-aktiviteten vha. en `SupportFragmentManager` erstatte det fragment, der viser `listView`et af alle ingredienser i køleskabet, med et nyt fragment (`fragment_display_ingredient`). Dette fragment lægger sig ovenpå `listView`et og samtidig låses layoutet bagved. `Fragment_Display_Ingredient` inflater her et nyt layout (`fragment_display_layout`). Her har brugeren mulighed for at indtaste informationer i de forskellige tekstfelter. I tekstfeltet for "Expire" er der tilføjet en `onClick`Listener, som åbner en `DatePicker` i et nyt fragment for at sætte datoen.

Herefter har brugeren to alternativer:

1. Brugeren kan vælge at "Cancel". Her vil `fragment_Display_Ingredient` kalde tilbage til Ingredient-aktiviteten vha. `getActivity()`, for at kunne bruge `SupportFragmentManager` til at fjerne fragmentet. På denne måde bliver der ikke gemt noget data, men fragmentet bliver blot fjernet.
2. Brugeren kan vælge "Save Ingredient", hvor metoden `setIngredientData()` bliver kaldt. Her oprettes der et bundle, hvor informationerne fra tekstfelterne i fragmentet bliver lagt i for at de kan gemmes i databasen. Det sker vha. interface-metoden `addIngredient()`. Her bliver dataen både gemt i databasen og `listView`et bliver opdateret med det nye element.

### RecipeList Activity



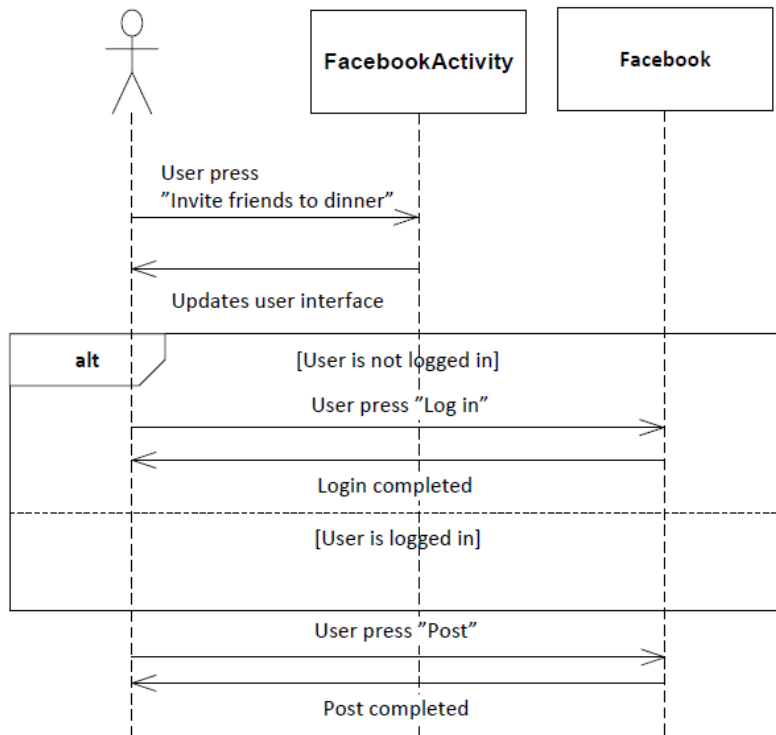
Figur 9 - Sekvensdiagram over RecipeList activity

Når brugeren trykker på "Find recipes" hentes opskrifter ud fra databasen. For hver ingrediens i køleskabet bliver opskrifterne gennemgået for matchene opskrifter til den ingrediens. Der laves altså en liste af opskrifter ud fra hver ingrediens. Dernæst sammenlignes alle opskriftlisterne og samles til en liste sorteret efter opskrifter med flest matchene ingredienser målt i procent.

Brugeren får nu vist listen af sorterede opskrifter, hvorfra han kan vælge en opskrift han ønsker at lave. Når en opskrift vælges åbnes opskriften og brugeren kan se hvilke ingredienser han allerede har og hvilke han mangler, samt en beskrivelse af opskriften.

### Facebook activity

Denne aktivitet afhænger af at brugeren har været inde og finde en specifik opskrift, som man vil invitere gæster til.



Figur 10 - Sekvensdiagram over Facebook activity

Brugeren trykker på "Invite friends to dinner", som starter aktiviteten FacebookActivity. Her vises der et nyt layout med et facebook-ProfilePictureView og en facebook-LoginButton. Derudover er der også en "Post"-knap. I FacebookActivity oprettes der en LoginManager, som er til at når brugeren skal logge ind. Der åbnes en dialog, hvor brugeren så kan logge ind. Når brugeren er logget ind vises brugerens profilbillede i ProfilePictureViewet. Herefter kan brugeren trykke på "Post". Vha. postStatusUpdate() tilføjes den valgte opskriffs navn til det objekt der skal postes. Herefter vises der en shareDialog, som er styret af den facebookAPI, som er importeret.

## Konklusion

Gruppen er meget tilfredse med den app vi har udviklet igennem kurset ITSMAP F15. Appen indeholder mange forskellige elementer af android programmering og har medvirket til at gruppen har fået en bred og god forståelse for mulighederne indenfor udviklingen af android applikationer. Selvom vi er tilfredse med appen er der stadig plads til en masse forbedringer, som evt. Kan implementeres på et senere tidspunkt.

- Direkte link fra notifikation til den varer der er for gammel's redigerings fragment.
- Mulighed for at tilføje egne opskrifter
- Oprette events direkte på Facebook i stedet for vægopslag
- Ikke optimalt at hente alle opskrifter ud hver gang der køres "Find recipe"-algoritmen

Arbejdsfordelingen har været flyvsk i gruppen. Derfor har alle haft en finger med i de største dele af projektet.

## Arbejdsfordeling

For at beskrive arbejdsfordelingen i projektet er der lavet nedenstående tabel. Grøn betyder man har haft stor del af arbejdsbyrden mens gul betyder mindre, men man har stadig været med inde over tingene. Af arbejdsopgaver er der taget udgangspunkt i appens use cases.

	Karsten	Rasmus	Nikolaj	Andreas
Add ingredient				
Edit ingredient				
Delete ingredient				
Inspect fridge				
Suggest recipes				
Invite friends to dinner				