

# Målgruppe

Problemområdet for dette projekt er madspild, der opstår på grund af manglende kreativitet til hvordan rester kan anvendes. Der er ingen der er i tvivl om, at der findes madspild rigtig mange steder. I erhvervslivet findes madspild i form af slagteren, der ikke når at få solgt sit kød, Nettos mælk der er ved at overskride sidste salgsdato, og plejehjemmet, der fik lavet for meget suppe til de ældre. I det private køkken sker der også madspild. I den store familie, købes der stort ind, og ikke alle fødevarer når at blive anvendt, enten fordi man simpelthen har købt for stort ind, eller også fordi fødevarerne ikke har passeret godt nok sammen. Man har måske haft lidt kylling liggende, og ikke syntes dette ville være passende i en lasagne. I en familie kan der være mange forskellige madønsker og kræsenheder. Oven i købet er der forskel på hvornår man bliver sulten, så samlet set er det ikke svært at forestille sig at lidt mad går til spilde fordi madlavningen ikke helt gik som tidligere planlagt, for eksempel under indkøbet.

Med alle disse steder man kan forestille sig at møde madspild, er det nødvendigt for os at tage en beslutning om hvilken gruppe vi vil fokusere på at mindske madspildet hos. En for stor målgruppe kan risikere at blive så uhomogen, at et system til målgruppen i sidste ende ikke vil kunne udvikles, fordi kravene til systemet vil være alt for forskellige.

For at indsnævre målgruppen vælger vi at fokusere på madspild i det private køkken. Det gør vi fordi de valgte informanter arbejder ikke med mad inden for erhvervslivet og kan dermed ikke kommentere på madspildet i det område.

I det private køkken, sker det til tider, at der smides mad væk. Årligt har man beregnet at hver dansker smider ca. 63 kg fødevarer væk. <http://politiken.dk/mad/madnyt/ECE527771/hver-dansker-smider-63-kg-mad-ud/>

En del af madspildet kan for eksempel ske efter endt madlavning, hvor man kan risikere at stå tilbage med nogle fødevarer, der helst skal bruges hurtigst muligt for ikke at blive for gamle. Man kunne for eksempel have lavet hele kyllingbryst med persille, og fået en masse af begge dele tilovers. Næste aften har man ikke lyst til at få samme ret, men maden kan ikke holde sig meget længere, så man fristes derfor til at smide maden væk. Maden kan dog godt anvendes på en anden måde end sidst. Man kunne for eksempel skære kyllingen i tern i en sammenkogt ret og persillen kunne komme i en salat i stedet for ud over kyllingen som i sidst. Denne viden er det ikke alle, der bærer rundt på, og hvis man med en kreativ tilgang blot prøver at blande nogle forskellige ingredienser, kan man ende med et måltid, der smager knap så godt som man forestillede sig.

# Systemdefinition(er):

## Systemdefinition 1 - online opskriftsregister:

Systemet skal fungere som et online opskriftsregister, der giver brugeren idéer til opskrifter han rent faktisk kan lave ud fra de madvarer han har. Systemet fokuserer på at mindske madspild, da forbrugere smider mad ud på grund af et manglende formål med anvendelsen af resterne. Brugere af programmet er en del af en husholdning og vil have meget varierende erfaringer inden for brug af internettet. Udviklerne af systemet er ulønnede studerende. Deadline for det færdige system kan ikke ændres. Systemet skal køre på en server, der kan tilgås via en webapplikation fra en internetbrowser på enhver type computer. På baggrund af en mængde fødevarer som input, findes forskellige opskrifter, der bedst muligt matcher disse. Opskrifterne skal kunne sorteres på flere måder, og ingredienser skal kunne huskes til næste gang, hvis ønsket.

## Systemdefinition 2 - madplanlægger:

Systemet fungerer som et planlægningsværktøj, som har til formål at planlægge madlavningen over en given tidsperiode (eks. dage, uger, måneder) ud fra de opskrifter, der er blevet lavet over de sidste par dage. Formålet med planlægningsværktøjet er at sikre, at brugeren får en sund og varieret kost igennem udvalgte opskrifter, hvor der også tages højde for ingrediensers vitaminindhold. Brugere af programmet vil være husholdninger, der har varierende erfaringer inden for brug af internettet. Udviklerne af systemet er ulønnede studerende. Systemet skal køre på en server, der kan tilgås via en webapplikation fra en internetbrowser på enhver type computer.

## BATOFF:

Systemdefinitionen er blevet formuleret ved hjælp af BATOFF-modellen. Modellen indholder 6 punkter, som er listet herunder:

- **Betingelser:** Betingelserne for systemets udvikling og brug.
- **Anvendelsesområde:** De dele af en organisation, som administrerer, overvåger eller styrer et problemområde.
- **Teknologier:** Den teknologi, som systemet udvikles til og ved hjælp af.
- **Objekter:** De væsentligste objekter i et problemområde.
- **Funktioner:** De systemdefinitioner, som understøtter arbejdsopgaver i anvendelsesområdet.

- **Filosofi:** Den filosofi, der ligger bag IT-systemets anvendelse.

Modellen er praktisk at anvende i forhold til at formulere systemdefinitioner, idet den sikre at systemdefinitionerne indholder en række væsentlige punkter. Ofte kan det også være svært at komme i gang med at formulere en systemdefinition. Her hjælper BATOFF-modellen, da man blot skal udfylde punkter, og man derefter ud fra de udfyldte punkter kan skrive en systemdefinition. Systemdefinition 1 og 2 er udformet på denne måde.

	<b>Systemdefinition 1</b>	<b>Systemdefinition 2</b>
<b>Betingelser</b>	Udviklere er ulønnede	Udviklere er ulønnede
<b>Anvendelsesområde</b>	Husholdning	Husholdning
<b>Teknologier</b>	Internetopkoblet PC / tablet / mobil	Internetopkoblet PC / tablet/ mobil
<b>Objekter</b>	Opskrifter Ingredienser	Opskrifter Ingredienser Vitaminer
<b>Funktioner</b>	Find opskrifter indeholdende valgter ingredienser	Planlægningsværktøj. Planlægge madlavning ud fra tidligere dages retter
<b>Filosofi</b>	Folk smider mad ud fordi de mangler et sted at bruge deres rester	Folk spiser ikke varieret nok, hvilket er en trussel for folkesundheden.

## Valg af systemdefinition

Informanterne kunne rigtig godt lide idéen om at søge på opskrifter ud fra de ingredienser man har i køleskabet. De kom med det samme med mange forslag til funktionaliteter et sådan system skal have (se møde 1). Når det kom til spørgsmålet om en madplan, virkede de knap så entusiastiske. De ville muligvis benytte en sådan, hvis den fokuserede på varieret kost (se møde 2). Denne funktionalitet er anderledes fra behovet om mindsket madspild, så derfor blev systemdefinition 2 lavet for at afspejle et madplansystem med fokus på varieret kost. Informanterne valgte dog systemdefinition 1, der mindsker madspild ved at præsentere opskrifter ud fra en mængde indtastede ingredienser. En madplan kan også være med til at mindske madspild, men vi ser ikke muligheden for, på en simpel måde, at integrere en

madplan med det eksisterende system. Vi kan kun se det som 2 radikalt forskellige systemer, og vælger derfor at fokusere på ét system, valgt på baggrund af informanternes ønsker, nemlig systemdefinition 1.

## **Kandidater til klasser og hændelser**

### **Klassekandidater**

#### **Negativer:**

##### **Person -**

Vi vælger at fjerne person-klassen, fordi vi mener, at vores løsning ikke skal være noget socialt media, og derfor mener vi, at det er selve husholdningen, der er fælles om madlavningen selvom det måske blot er en person, der laver mad og står for indkøb og lignende. Brugeren af programmet er ikke en del af problemområdet og skal derfor ikke modelleres som klasse.

##### **Køkken -**

Køkken og husholdning dækker over samme del af problemområdet, og vi fjerner derfor køkken og vurderer bagefter om husholdning skal være en klasse.

##### **Husholdning -**

En husholdning repræsenterer et hjem, som indeholder én til flere personer. Det er ikke en del af problemområdet at holde styr på eller at kommunikere med andre husstande.

##### **Køleskab/skab/opbevaringsskab -**

Om råvarerne befinder sig i et køleskab eller i en skuffe er, for os, uinteressant, derfor er vi ikke interesseret i at modellere disse skabe som en klasse.

##### **Køkkenredskab/komfur -**

Ligesom ved køleskab/anden opbevaring er vi ikke interesseret i at modellere hvilke redskaber husholdningen har adgang til.

##### **Service -**

Vi har valgt at fjerne klassen service, fordi denne ikke findes i vores problemområde. Service er noget, der bliver brugt, når man er i færd med at spise det færdige mad, og ikke under selve madlavningen.

##### **Butik -**

Vi har valgt at fjerne klassen Butik, da vores system fokuserer på madspild og varieret kost. En modellering af butikker ville være relevant, hvis vores fokus lå på at begrænse udgifter på mad, men da dette imidlertidig ikke er situationen i problemområdet, bliver den udeladt.

### **Typisk/atypisk ingrediens -**

Det er ikke en del af problemområdet at folk ikke er klar over hvilke ingredienser der er normale/unormale at have.

### **Enhed/Mængde -**

Enhed og mængde er ikke klasser, men attributter til en ingrediens.

### **Madplan -**

På baggrund af diskussionsafsnittet omkring de to systemdefinitioner, vælger vi at fjerne klassen madplan.

### **Positiver:**

#### **Ingrediens +**

I opskrifter bruges der flere ingredienser. En ingrediens består af en råvare og en mængde af denne. Det er et problem at finde opskrifter, der indeholder ingredienser svarende til de råvarer man har til rådighed. Det er et problem for informanterne at maden laves i for store portioner, altså er det et problem hvis en opskriffs ingredienser indeholder store mængder.

#### **Bogmærke +**

Det er en del af problemområdet, for brugere at huske de gode opskrifter. Der vil til tider blive benyttet en opskrift, der er så god, at den er værd at gemme til en anden gang. Derfor beholder vi denne klasse.

#### **Råvare +**

En råvare findes i køleskabe og på madhylderne i husholdningerne. Det er et problem at finde opskrifter, der kun indeholder disse råvarer, derfor skelnes der mellem ingredienser og råvarer.

#### **Indkøbsliste +**

Vi vurderer, at der i en husholdning ofte bliver skrevet en indkøbsliste med de ting man mangler. Indkøbslisten kan være skrevet på baggrund af en opskrift man gerne vil lave, eller en hel madplan man gerne vil følge over en længere periode.

## **Hændelseskandidater:**

### **Valgte hændelser**

- Råvare opbrugt
- Råvare smidt ud
- Råvare købt
- Opskrift fundet
- Opskrift valgt
- Opskrift smidt ud
- Bogmærke tilføjet

- Bogmærke fjernet
- Indkøbsliste oprettet
- Indkøbsliste færdig
- Indskøbsliste smidt ud
- Ingrediens tilføjet
- Ingrediens fjernet

## Fravalgte hændelser

Følgende hændelskandidater er blevet fravalgt, da de enten hørte til nogle klasser, som er blevet fravalgt, eller ikke er relevante nok, for de valgte klasser:

- Køkkenredskab benyttet (systemet skal ikke holde styr på køkkenredskaber)
- Råvare benyttet (systemet skal ikke holde styr på mængden af råvarer hos brugeren)
- Mæthed opnået (fra fravalgte klasser: bruger, person)
- Madrest opstået (systemet skal ikke behandle råvarer forskelligt om det er rester eller ej)
- Bord opdækket (fra fravalgt klasse: husholdning)
- Opvask taget (fra fravalgt klasse: køkken)
- Service benyttet (fra fravalgt klasse: service)
- Køleskab åbnet (fra fravalgt klasse: opbevaringsskab)
- Køleskab lukket (fra fravalgt klasse: opbevaringsskab)
- Opskrift vurderet (opskrift valgt indebærer, at man har vurderet opskriften)
- Opskrift anmeldt (ikke en del af problemområdet at anmelde opskrifter)
- Sult opstået (fra fravalgte klasser: bruger, person)
- Madlavning afsluttet (fra fravalgte klasser: bruger, person)
- Madlavning påbegyndt (fra fravalgte klasser: bruger, person)
- Råvare identificeret (overvåges i form af "råvare købt"-hændelsen med samme resultat)
- Ingrediens identificeret (indgår i hændelsen opskrift valgt)
- Madplan lagt (fra fravalgt klasse: Madplan)
- Madplan startet (fra fravalgt klasse: Madplan)
- Madplan afsluttet (fra fravalgt klasse: Madplan)
- Opskrift fravalgt (fra fravalgt klasse: Madplan)

## Brug

Aktører (kandidater)

- Bruger
- Administrator
- Crawler

Brugsmønstre (kandidater)

Fravalgte:

- Skalering -
  - Skalering af ingredienser i opskrifter i forhold til antallet af personer
  - Fjernet, da skalering indgår i søgningen fordi der er ingen grund til at skalere opskriften i hver visning og kan dermed indgå som en slags "filter" i søgningen.
- Råvarehåndtering -
  - Når man gemmer ingredienser i opskrifter... (?)
  - Fjernet, da råvarehåndtering indgår i søgningen

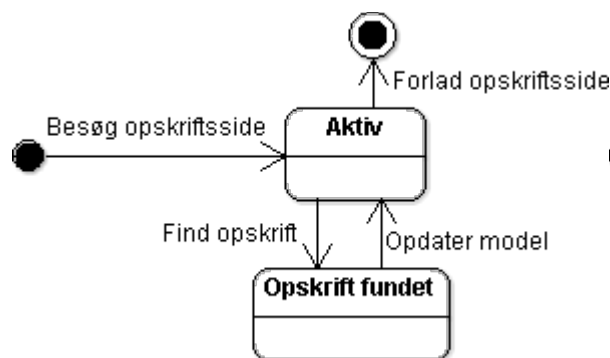
- Overvågning -
  - Hvad skal vi egentligt overvåge?
- Begrænsning -
  - Fx. ingen kød, glutenfri
  - Fjernet, da begrænsning indgår i søgningen
- Sortering -
  - Sortere opskrifter i forhold til forskellige filtre
  - Fjernet, da sortering indgår i søgningen
- Madplanlægning -
  - Fjernet, da madplanen blev fravalgt i problemområdet

Valgte:

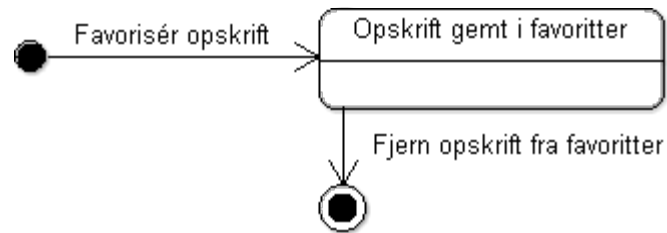
- Søgning
  - At søge på opskrifter i forhold til de indtastede ingredienser
- Favorisering
  - At favorisere/bogmærke en opskrift
- Indkøbslistehåndtering
  - Skal kunne tilføje ingredienser fra flere forskellige opskrifter på én fælles indkøbsliste
- Synkronisering
  - I stedet for et loginsystem, vil vi benytte cookies og derfor skal synkronisering mellem forskellige enheder håndteres
- Anmeld
- Fejlhåndtering
- Crawling
  - Når crawleren besøger opskriftshjemmesider og tilføjer/opdaterer/sletter opskrifter til/fra Foodl's indeks

## Brugsmønstre

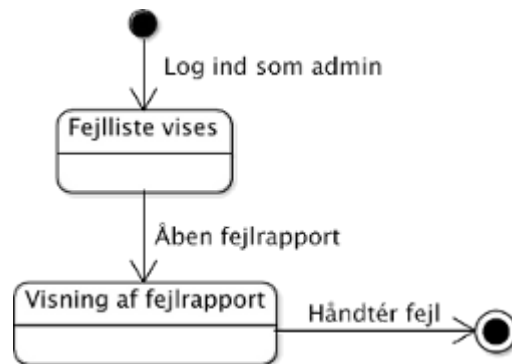
Crawling:



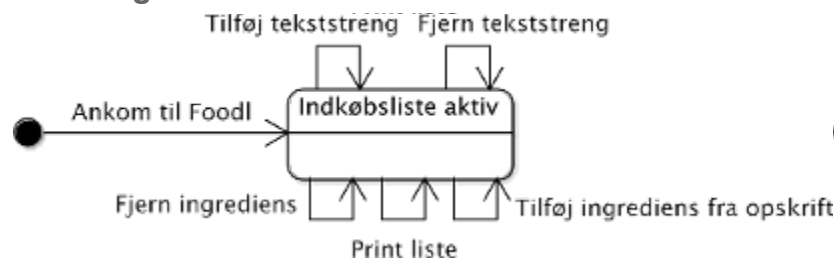
### Favorisering:



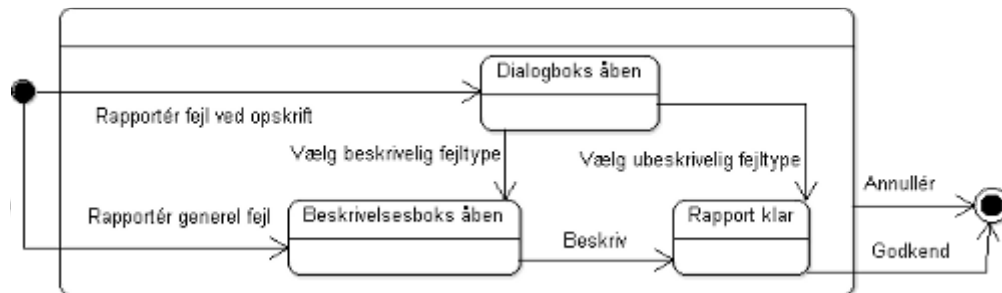
### Fejlhåndtering:



### Indkøbslistehåndtering:

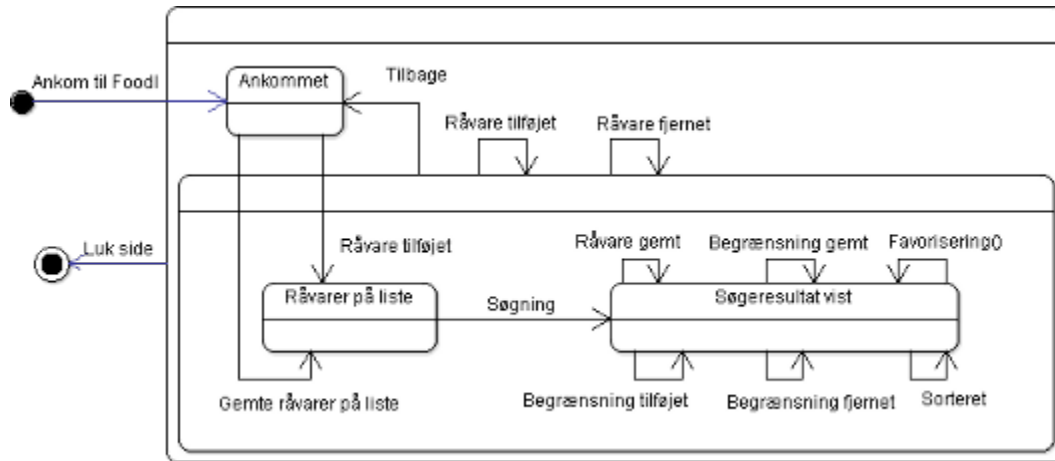


### Rapportering:

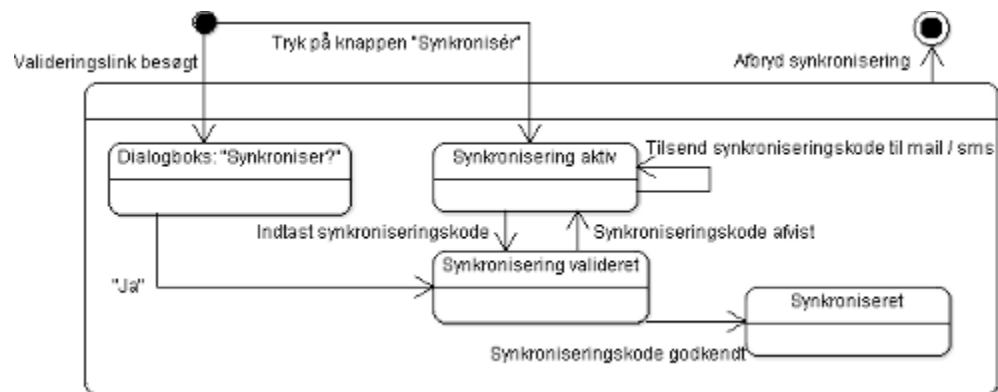




## Søgning:



## Synkronisering:



## Brugsmønsterspecifikationer

<i>Favorisering</i>
<b>Brugsmønster</b> Favorisering igangsættes af <i>brugeren</i> . Når <i>brugeren</i> har lavet en søgning, og flere forskellige opskrifter vises som søgeresultater, kan <i>brugeren</i> klikke på favoriser-knappen tilhørende den enkelte opskrift, for at favorisere denne. Når en opskrift er blevet favoriseret, tilføjes denne til en liste af favoritopskrifter. Det er muligt at fjerne opskriften fra favoritlisten på to måder. Enten fra samme sted, som favoriseringen blev tilføjet, eller direkte i favoritlisten.
<b>Objekter</b>
<b>Funktioner</b>

<i>Rapportering</i>
<b>Brugsmønster</b> Rapportering igangsættes af <i>brugeren</i> , når denne opdager en fejl på hjemmesiden. Hvis <i>brugeren</i> opdager en fejl, der har med et søgningsresultat (opskrift) at gøre, så klikkes der på en rapporteringsknap, der er ved det enkelte søgningsresultatet. Når der skal rapporteres en fejl vedrørende opskrifter, så åbnes en dialogboks på siden, hvor <i>brugeren</i> herefter skal vælge en fejltype. Der skelnes mellem beskrivelige og ubeskrivelige fejltyper. Et eksempel på en ubeskrivelig fejltype er bl.a., hvis et link ikke fungerer, som hører under fejltypen "dødt link". De ubeskrivelige fejltyper behøver ingen beskrivelse, da fejltypen er beskrivelse nok i sig selv. Vælger <i>brugeren</i> derimod en beskrivelig fejltype, så præsenteres en beskrivelsesboks for <i>brugeren</i> , hvor fejlen beskrives med tekst. Derefter er rapporten klar, og <i>brugeren</i> skal nu godkende rapporten, inden den bliver sendt til <i>administratoren</i> . Derudover er der en generel rapporteringsknap, der vedrører andre, generelle fejl på siden. Når denne knap benyttes, så dirigeres <i>brugeren</i> direkte hen til en beskrivelsesboks, hvor fejlen beskrives med tekst. Til slut skal rapporten godkendes af <i>brugeren</i> , inden den bliver sendt til <i>administratoren</i> . Det er altid muligt at annullere rapporteringen under alle tilstande i brugsmønstret.
<b>Objekter</b>
<b>Funktioner</b>

### ***Fejlhåndtering***

#### **Brugsmønster**

Fejlhåndtering igangsættes af *administratoren*. *Administratoren* logger ind på hjemmesiden, og bevæger sig ind på fejlhåndteringssiden. Her præsenteres en liste af fejlrapporter, der via *brugeren* er blevet rapporteret og dokumenteret i systemet. *Administratoren* kan derefter klikke på en fejlrapport i listen for at se en detaljeret beskrivelse af den givne fejl, som derefter kan håndteres.

#### **Objekter**

#### **Funktioner**

### ***Indkøbslistehåndtering***

#### **Brugsmønster**

Indkøbslisterhåndtering aktiveres så snart *brugeren* ankommer til Foodl's forside. Her kan *brugeren* tilføje og fjerne ingredienser til/fra indkøbslisten så meget *brugeren* vil. Dette kan Både ske fra Foodl's forside, og fra søgningsresultatet (en liste af opskrifter). Desuden kan *brugeren* tilføje og fjerne tekststreng, så det også er muligt at tilføje ikke-ingredienter, dvs. madvarer uden tilhørende enheder, eller ikke-råvarer (eksempelvis "toiletpapir"). *Brugeren* har også mulighed for at printe indkøbslisten ud, så denne kan medbringes, når der skal handles ind. Indkøbslistehåndtering holdes altid aktiv.

#### **Objekter**

#### **Funktioner**

### ***Synkronisering***

#### **Brugsmønster**

Synkronisering igangsættes af *brugeren* ved at klikke på knappen "synkronisér" eller ved at klikke på et synkroniseringslink, som *brugeren* har modtaget med mail. Hvis *brugeren* klikker på knappen "synkronisér", så præsenteres *brugeren* for en side, der giver *brugeren* mulighed for at synkronisere andre computere med *brugers* nuværende session. En synkroniseringskode genereres automatisk og vises, og det er denne kode, der skal indtastes på de andre computere for at synkronisere den nuværende session. *Brugeren* kan vælge at få tilsendt et synkroniseringslink, der er en erstatning for synkroniseringskoden. Klikker man på et synkroniseringslink præsenteres *brugeren* for en dialogboks med mulighed for at acceptere

synkroniseringen. Accepterer *brugeren* synkroniseringen, valideres synkroniseringen, hvorefter synkroniseringen igangsættes. Hvis koden ikke bliver valideret, så bliver *brugeren* sendt tilbage til siden, hvor synkroniseringskoden skal indtastes igen. På et vilkårligt tidspunkt i processen er det muligt at afbryde synkroniseringen og komme tilbage til forsiden.

#### Objekter

#### Funktioner

### Søgning

#### Brugsmønster

En søgning igangsættes af *brugeren*, ved at indtaste et antal forskellige råvarer og derefter trykke på "søg". En del af de indtastede råvarer kan også være gemte ingredienser fra tidligere søgninger. Efter en søgning, vises en mængde opskrifter baseret på de råvarer, der er blevet indtastet. Det er muligt at tilføje eller fjerne råvarer, der kan yderligere specificer søgningen. Opskrifterne sorteres i første omgang efter, hvor godt deres ingredienser matcher de valgte råvarer. *Brugeren* kan vælge en sekundær sortering, hvor *brugeren* har mulighed for at sortere efter bedømmelse eller navn. Opskrifterne kan også filtreres på flere måder (også samtidig), så der kun vises opskrifter uden fx kød, svin og/eller gluten. I søgeresultatet er det muligt at tilføje samt gemme og fjerne begrænsninger og råvarer. Søgningen kan under alle tilstande afsluttes ved at lukke siden.

#### Objekter

#### Funktioner

### Crawling

#### Brugsmønster

Crawling igangsættes af *crawleren*. Under crawling besøges alle opskrifter på en opskriftsside. For hver opskrift undersøges det om opskriften allerede eksisterer. Hvis den eksisterer, kontrolleres opskriften for ændringer. I tilfælde af ændringer, opdateres den eksisterende opskrift i systemets model, ellers springes opskriften over. En ny opskrift tilføjes altid.

#### Objekter

#### Funktioner

## Aktørtabel

	<b>Brugsmønstre</b>			
<b>Aktører</b>		<i>Bruger</i>	<i>Administrator</i>	<i>Crawler</i>
	<i>Søgning</i>	✓		
	<i>Favorisering</i>	✓		
	<i>Indkøbslistehåndtering</i>	✓		
	<i>Synkronisering</i>	✓		
	<i>Fejlhåndtering</i>		✓	
	<i>Rapportering</i>	✓	✓	
	<i>Crawling</i>			✓

# Funktioner

## Funktionsliste

*Funktionslisten er ikke helt færdig*

Navn	Funktionstype	Kompleksitet
Find opskrift	Beregning	Simpel
Opdater opskrift	Opdatering	Simpel
Opret opskrift	Opdatering	Simpel
Fjern opskrift	Opdatering	Simpel
Opret favorit	Opdatering	Simpel
Fjern favorit	Opdatering	Simpel
Opret indkøbsliste	Opdatering	Simpel
Fjern indkøbsliste	Opdatering	Simpel
Tilføj ingrediens (indkøbsliste)	Opdatering	Simpel
Fjern ingrediens (indkøbsliste)	Opdatering	Simpel
Søg	Beregning	Kompleks
Skaler søgeresultat	Beregning	Simpel
Hent synkroniseringskode	Aflæsning	Simpel
Indtast synkroniseringskode	Opdatering	Medium
Vis favoritter	Aflæsning	Simpel
Opret fejlrapport	Opdatering	Medium
Luk fejlrapport	Opdatering	Simpel
Print indkøbsliste	Aflæsning	Simpel

Se indkøbsliste	Aflæsning	Simpel
Tilføj streng (indkøbsliste)	Opdatering	Simpel
Fjern streng (indkøbsliste)	Opdatering	Simpel
Start crawl af opskriftside	Beregning	Kompleks
Afslut crawl af opskriftside	Beregning	Simpel