# Design og implementering

## Web app

Figur - WebApp mappestruktur. MVC pattern

### Design

Mappestrukturen (Figur 1) i web applikations projektet er generet af Visual Studio under oprettelsen af projektet. Den følger MVC mønstret og har derfor mapperne: Controllers, Views og Models. Models mappen er i dette projekt gjort overflødigt da de klasser der burde ligge i denne mappe ligger i et class library ved navn: ”SmartFride\_WebModels”, som har taget udgangspunkt i modellaget fra den anden applikation.

App\_Data mappen bruges ikke i dette projekt da al data der skal gemmes i denne applikation gemmes i en database.

App\_Start indeholder klasser der henter scripts og bestemmer hvilken side der skal vises under opstart af applikationen. Disse er autogenerede og ikke ændret i.

Content mappen indeholder de .cs stylesheet, som hvert view bruger for at style siderne.

Images, fonts og Scripts mapperne indeholder henholdsvis de billeder, de scripts og de fonts, som bruges i applikationen.

I Web.config (XML fil) indeholder information om hvilke assamblys applikationen skal hente, hvilke connectionsstrings den skal bruge og meget mere.

Global.asax (cs fil) kører de filer der ligger i App\_Start under opstart af applikationen.

Hver controller har et View tilknyttet, hvor controlleren sørger for funktionalitet og Viewet sørger for at rendere html kode, som browseren kan læse og brugeren herved kan se i sin browser.

Dette projekt består af 4 controllers:

* HomeCotroller
* LisViewController
* AddItemController
* EditItemController

Homecontroller’en sørger for applikationens forside og har viewet ’Index.cshtml’ tilknyttet. Controlleren eneste funktionalitet er at returnere Index viewet.

LisViewController’en henter seneste data fra databasen og viser denne data i et grid. I dette view som denne controller returnere, har brugeren mulighed for at tilføje, redigere og slette items. Hvis ”tilføj item” vælges, kaldes AddItemController’en. Vælges ”redigér item” kaldes EditItemController’en og det item der skal redigeres sendes med. ”Slet item” kalder en funktion i LisViewController’en, som sletter det valgte item og returnere LisView’et med den nye opdateret model.

### Implementering

#### Controllers

#### Views

#### Models

### Test

**Design:**

Her beskrives det mere detaljerede design for hhv. HW og SW.

**Implementering:**

Her beskrives implementeringen i detaljer (kredsløbsberegninger, klassediagrammer, sekvensdiagrammer, state-diagrammer, aktivitetsdiagrammer. sourcekode udsnit m.m.).

**Test:**

Her beskrives de for projektet gennemførte enhedstests. Integrationstests og accepttest kommer i de følgende afsnit.

Evt. testprogrammer vedlægges på CD-ROM.