Grupp 2 Adam Clark, Dag Fridén, Fredrik Lundberg, Oscar Blomqvist

Dokumentation

Detta dokument är en dokumentation för det projektarbete som genomfördes under september och oktober månad i kursen Web Applikationer (DAT076) vid Chalmers Tekniska Högskola

Projektarbete

DAT076

Innehåll

[Projektmedlemmar 1](#_Toc370242180)

[Introduktion 2](#_Toc370242181)

[Förutsättningar 3](#_Toc370242182)

[Användarfall 3](#_Toc370242183)

[Register 4](#_Toc370242184)

[Login 5](#_Toc370242185)

[Add items 6](#_Toc370242186)

[Add list 7](#_Toc370242187)

[Logout 8](#_Toc370242188)

[Fysisk uppsättning 9](#_Toc370242189)

[Participating software components 9](#_Toc370242190)

[The modules (packages) 9](#_Toc370242191)

[A layered view of the application 9](#_Toc370242192)

[The object oriented model as an UML class diagram 9](#_Toc370242193)

[Classes central to understand the application 9](#_Toc370242194)

[Iff needed comments for methods or statements (ska egentligen gå att förstå koden) 9](#_Toc370242195)

# 

# Projektmedlemmar

Adam Clark [adam.clark@xenit.se](mailto:adam.clark@xenit.se) 911129-1838

Dag Fridén [dag@friden.com](mailto:dag@friden.com) 910122-2355

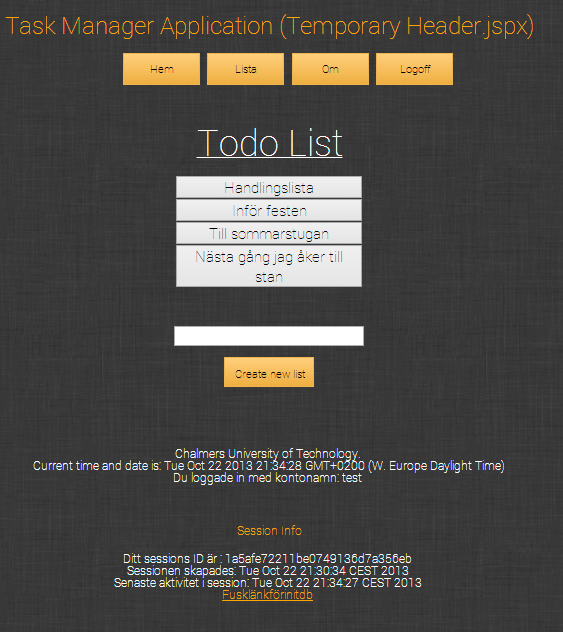
Fredrik Lundberg [lundberf@student.chalmers.se](mailto:lundberf@student.chalmers.se) 881110-4937

Oscar Blomqvist [boscar@student.chalmers.se](mailto:boscar@student.chalmers.se) 880228-5034

Översikt

# Introduktion

Denna applikation är skapad för att göra olika att-göra listor. Några exempel på detta är en inköpslista för mat, olika uppgifter man ska göra, kom-ihåg lista till en fest och så vidare.

Hemsidan är uppbyggd på ett sätt som möjliggör inloggning med konto för att skapa privata listor vilka inte delas med övriga användare av applikationen. I en framtida vidareutveckling av applikationen kommer även delade och publika listor finnas med.

Listorna visas på ett intuitivt sätt på den webbsida där applikationen är publicerad. För att vidare göra applikationen skalbar och enkel att hantera ligger det i bakgrunden en databas vars primära uppgift är att spara listor och inloggningsinformation.

Applikationen har även stöd för att vidareutvecklas till befintliga och nya användargränssnitt. Detta innebär till exempel att applikationen skulle kunna köra parallellt på en mobil enhet samtidigt som den används i en webbläsare – med samma information på båda platser.

Applikationen har utvecklats av fyra studenter vid Chalmers Tekniska Högskola under höstterminen 2013 i kursen Web Applikationer, DAT076.

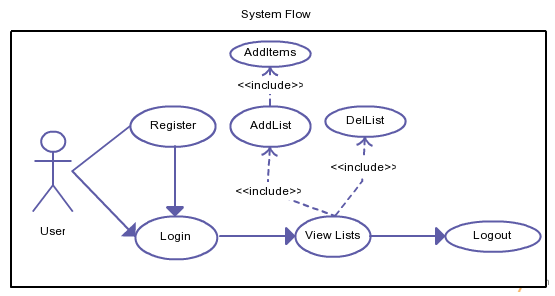
# Förutsättningar

Applikationen är modulärt uppbyggd med en så kallad frontend del samt en backend. Applikationen är byggd med Java Enterprise Edition och några av de övriga teknikerna/språken som använts är JavaScript (JQuery) och JavaDB (Derby).

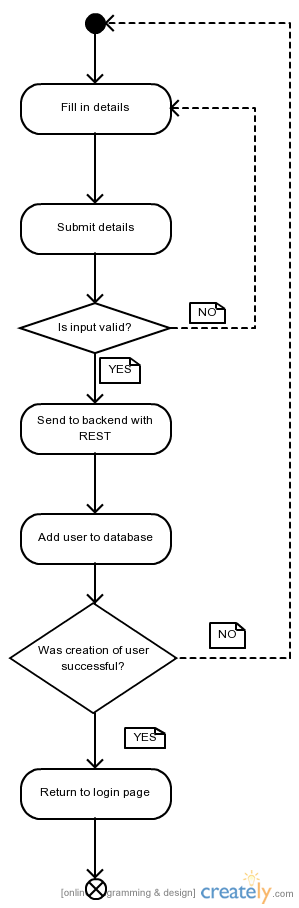
Under uppbyggnaden och testningen av applikationen är den körd på open-source applikationsservern Glassfish (CDDL/GPL).

# Användarfall

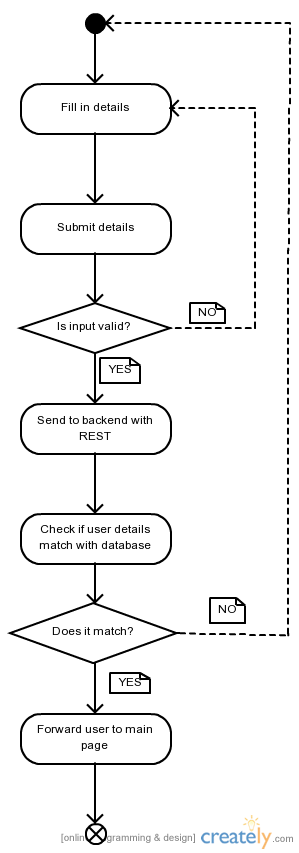
Nedan följer några av de användarfall som kan ske vid användandet av applikationen. Den första figuren förklarar olika user-case där varje cirkel innebär ett scenario vilket mer detaljerat förklaras i efterföljande figurer.



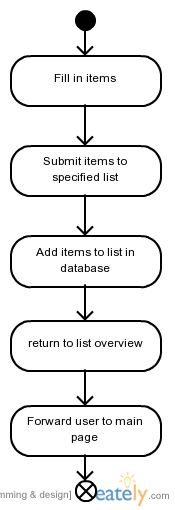
## Register



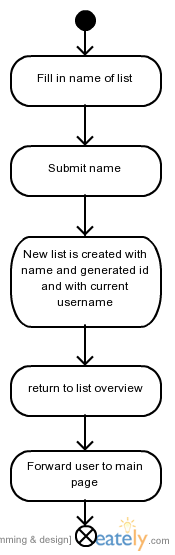
## Login



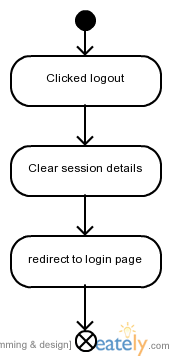
## Add items



## Add list



## Logout



Teknisk Design

# Fysisk uppsättning

Vid en eventuell utrullning av denna applikation så skulle det krävas en webbserver som kan hantera Java EE applikationer. En väl fungerande server är den som användes vid utvecklingen av applikatioen – GlassFish Server 3.1.2.2. Applikationen kräver en domän i Glassfish servern och inga speciella inställningar krävs för en grunduppsättning. Det finns många möjligheter att skala denna applikation med lastbalansering mellan olika servrar och till exempel lasta databasen mot ett databaskluster.

Det går att avlasta denna server genom att göra en migrering av databasen till en extern server. Detta är att rekommendera i det scenario då applikationen skulle rullas ut live.

# Komponenter

I denna applikation ingår många delkomponenter, de primära kommer i detta kapitel presenteras och en förklaring till i vilka delar av applikationen de används samt vilket syfte de används för.

## Maven

Maven är ett verktyg som används för att förenkla utveckling av Java-applikationer. Det används för att automatiskt bygga programmet och hämta de beroenden applikationen har. Maven använder sig av ett XML-dokument som heter POM.xml och det är i denna fil projektet beskrivs och vilka beroenden som applikationen har. Beroenden är olika paket och moduler som krävs för att applikationen ska fugnera. Till exempel används Maven för att hämta de beroenden som krävs för att applikationen skall kunna arbeta mot en databas.

Maven används både i den frontend och backend applikationen består av.

# The modules (packages)

# A layered view of the application

# The object oriented model as an UML class diagram

# Classes central to understand the application

Purpose/responsibility?

# If needed, comments for methods or statements (ska egentligen gå att förstå koden)