

## **Linjekoll dokumentation**

Jesper Josefsson (860409-5276), [jesper.josefsson@gmail.com](mailto:jesper.josefsson@gmail.com)

Linus Oleander (880613-4873), [linus@oleander.nu](mailto:linus@oleander.nu)

Johan Wickberg (840331-4878), [johanwi@student.chalmers.se](mailto:johanwi@student.chalmers.se)

## Översikt

Linjekoll är en webapplikation som visar kollektivtrafiknät i realtid. Siten ger en grafisk representation över valda linjer och visar var varje fordon, däribland spårvagn, längs linjen befinner sig just nu. Applikationen vänder sig till allmänheten.

## Användare

Siten har en inloggningsfunktion för vanliga användare. Administratörer kan justera flödet av data. Inloggningen är tänkt att ge tillgång till sparad data, såsom favoritinställningar.

## Use cases

1. Välja linje att titta på
2. Stänga ner en linje
3. Spara linje så den finns kvar vid nästa besök av sidan
4. Söka på hållplatser, få fram linjer för en viss hållplats
5. Få texten auto-completed när man söker efter hållplatser

## Design

Applikationen kan sägas bestå av två huvudsakliga delar:

Del 1 läser information från Västtrafiks server och omvandlar denna till en grafisk representation av spårvagnslinjer där spårvagnar rör sig som små cirklar. Denna del är uppbyggd med hjälp av diverse JavaScript-bibliotek som ger den en klientbaserad MVC-arkitektur.

Del 2 (lineart-enterprise-app) sköter registrering och login av användare och sparar användarinformation, såsom favoritlinjer. Denna del är uppbyggd som en Java EE-applikation.

### Tiers (Del 1)

Del 1 är svår att beskriva i termer av tiers eftersom hela MVC-modellen implementeras i klienten.

### Layers (Del 1)

View: jQuery template, underscore template, Backbone view, Raphaël-objekt

Controller: Backbone router

Model: Backbone model

Persistency service: local storage, cookies, websocket, Västtrafiks server, Redis in memory key value database

## **Packages (Del 1)**

Del 1 består av en rad underapplikationer som i ena änden hämtar data från Västtrafiks server och i andra änden presenterar det hela grafiskt. Däremellan sker flera steg som behandlas av dessa underapplikationer.

*Api-server-js*: Denna applikation som är skriven med hjälp av js-biblioteket node.js, tar emot data från Västtrafik och validerar den. Korrekt data släpps vidare.

*Socket-server*: Denna applikation är en websocketserver som skickar vidare data till front end och till en cache.

*Web-client*: Denna applikation agerar som front end och presenterar grafiskt den data som kommit från socket-server. För vektorgrafik används biblioteket raphaël.js och för mvc-modellering används backbone.js. Information från databasen, för t.ex. en linje, laddas enkelt in via ett REST-API byggt på vår nuvarande Java-stack.

## **Tiers (Del 2)**

Client side presentation: HTML5, CSS3

Server Side Presentation: JSF

Business logic: Entities (see nedan), Session beans (see nedan), Managed beans (see nedan)

EIS: MySQL server

## **Layers (Del 2)**

View: JSF-pages, CSS

Controller: Managed beans (see nedan), Servlet (see nedan)

Model: Session beans (see nedan), Entities (see nedan)

Persistency service: MySQL server

## **Packages (Del 2)**

### **Managed beans**

*se.linjekoll.beans.backing*: innehåller en backing bean som kommunicerar med en JSF-sida, vars input lagras i bönans variabler.

*se.linjekoll.beans.controller*: innehåller en controller bean vars metoder kan anropas av JSF-sidan, vilket i sin tur gör att metoder hos Authenticator-klassen anropas. Bönan använder sig av backing bean för att få tillgång till den bönans variabler.

*se.linjekoll.beans.session*: innehåller session beans som innehåller information om befintlig användarsession.

### **Servlet**

*se.linjekoll.web.servlets*: innehåller en servlet som hämtar information från databasen på begäran av klienten.

## Session beans

*se.linjekoll.authentication*: innehåller logik för autentiseringsmekanismen.

*se.linjekoll.persistence.facades*: innehåller fasader för entityklasserna.

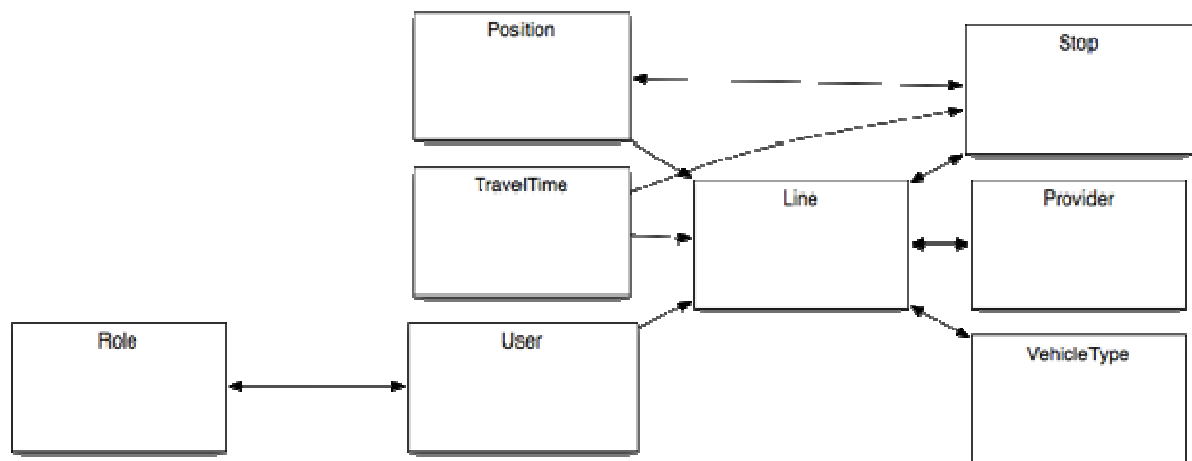
*se.linjekoll.persistence.utilities*: innehåller hjälpklasser för JSON-hantering.

## Entities

*se.linjekoll.persistence.entities*: innehåller entities.

*se.linjekoll.persistence.entities.superclasses*: innehåller en superklass för entities.

## UML-diagram



# Object Model - Linjekoll

## Viktiga klasser

### *Del 1*

Saknar javaklasser.

### *Del 2*

*AuthControllerBean* (se.linjekoll.beans.controller)

Sköter kommunikationen mellan web-applikationen och Authenticator i ejb-modulen.

### Authenticator (se.linjekoll.authentication)

Har metoder för att lägga till en användare i databasen, och för att genomföra autentisering.

## **Använda teknologier**

JSF

JavaScript

JPA

EJB

Enterprise App

AJAX

Web sockets

Köhantering

Primefaces

HTML5

CSS3

SASS

Dokumentbaserad database