Mind Map PIM

A-Cube-N

Department of Computer Science, University of Pretoria

May 20, 2016

Overview

Project Overview

Mind Mapped PIM is a system that will process existing PIM platforms such as Facebook and Gmail and construct a mind map using that data. The mind map will allow functionality of the sources e.g. send an email. It will be programmed in a way to display information that is most relevant to you at a particular time in your life.

What it is not

The Mind Map PIM is not a new social media platform. Rather it acts as a filter for your current social media platforms. It will use intelligent algorithms to determine what information you would most likely want to see.

Platforms we will use

We will make use of the following social media platforms by using the APIs that are provided by them:

1. Google services (priority)
    1. Gmail
    2. Calendar
    3. Notes
    4. Plus
2. Facebook
3. LinkedIn

We will be using dependency injection to implement the different platforms. This will enable future expansion to more platforms.

Software Architecture Overview

We will be using a layered system architecture to implement the system. This will allow us to better achieve:

- modularity
- flexibility
- scalability
- access limitation

Some Software Architecture Requirements

Usability Very simple to use since users with varying computer literacy will use the system.
Performance The system should be able to display information in real-time. New information should be displayed almost immediately.
Security Since we are dealing with sensitive information it is vital that the best encryption and secure connections are used when communicating with external systems.
Reliability The system will run seven days a week, twenty-four hours per day.

Software Architecture Design

Model-View-Controller (MVC): this gives us many benefits mainly:
Separation of design concerns: Because of the decoupling of presentation, control, and data persistence and behavior, the application becomes more exible; modifications to one component have minimal impact on other components.
More easily maintainable and extensible: Good structure can reduce code complexity. As such, code duplication is minimized.
Promotes division of labour: Developers with different skill sets are able to focus on their core skills and collaborate through clearly defined interfaces.

Technologies

Technologies we will be using:

- JavaEE
- TomEE (JavaEE version of Apache Tomcat)
- Google Parsey McParseFace (Natural Language Processor for English)

Software Architecture Persistence

For persistence we will be using MongoDB. This choice was made on the fact that it will integrate better with our OO system and also the fact that services like Mind Map PIM has the potential to attract many users. Benefits include:

- high write loads
- good for Big Data scenarios
- highly scalable
- document orientated storage

Software Architecture Interface

The system will be accessed via a web interface (our primary goal) and an Android app (as a secondary goal). The mind map will be presented in a fashion similar to the way that musicroamer.com displays related music artists (see next slide).

- At the center is the root node.
- Each social media platform has a node going out from the root node.
- Relevant information is displayed about each platform.
- User can expand any node to find relevant information about that node.
- The user can specify a ply depth to specify how many 'branches' the mind map should allow from a main node
- Should a node allow functionality e.g. Facebook node allows you to comment, this functionality will be provided in a panel to the right of the mind map

Software Architecture Interface

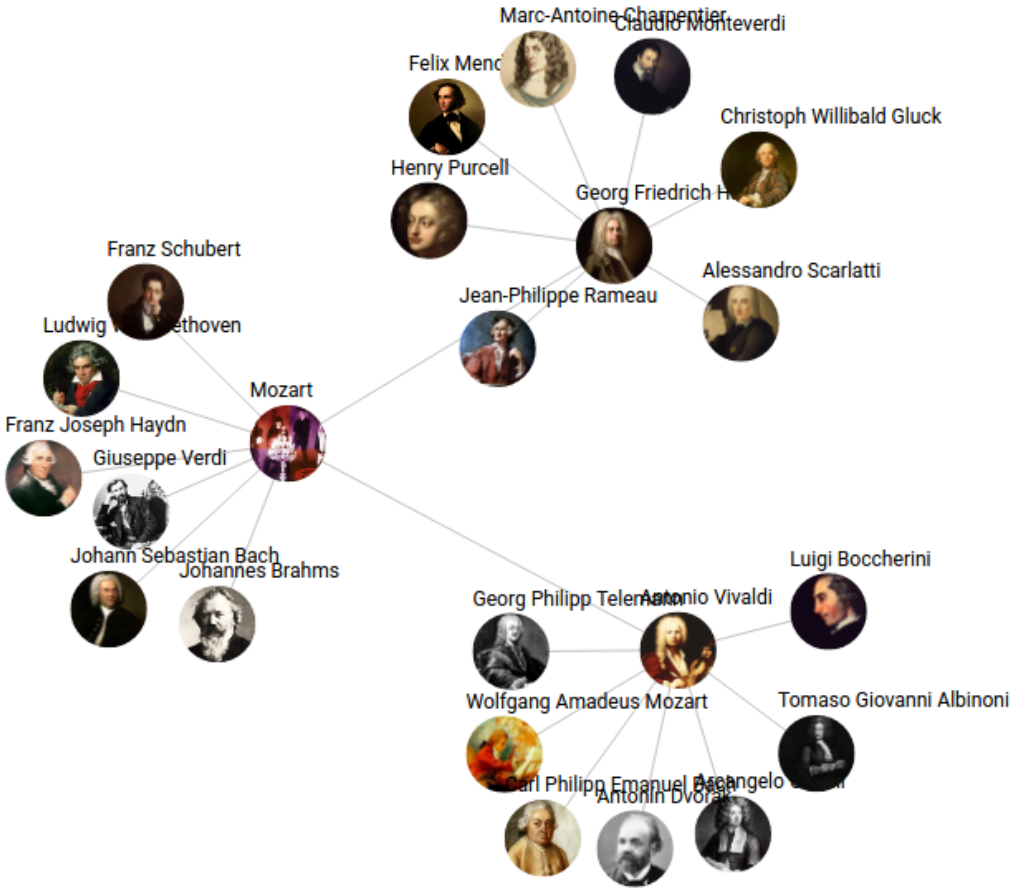

Figure : Music Roamer layout of a music mind map

Initial Design 1

Initial Design2

The End