Verein

Data Model

Added by Donald Kossmann on 4/15/2013 2:22 PM, last edited by Donald Kossmann on 4/15/2013 4:30 PM

Introduction

This document covers the core of the system. The core of the system is the "database backend" that is used to enter information and to implement basic user-facing services; e.g., applications that remind patient to take their medication. What this document does *not* cover are issues regarding the user interface, concrete applications, and analytics (e.g., how to create value from the data). Other documents must cover these technical aspects of HealthBank.

Overview

The core of the system has three main components:

a.) Feed: Just like Twitter, this data structure keeps all health records, events, etc. associated to a user.

b.) Circles: Just as in Google+, a user defines circles in order to share items of the feed with other users. There are three pre-defined circles: "friends", "family", "doctor", "science", and "commercial". The user enters other users into circles and decides which circles can see which items.

c.) Spaces: Items of a user are grouped into "spaces". Based on these spaces, applications and the user interface is built. There are three pre-defined spaces: "wellness", "fitness", and "health". Typically, the classification of an item to a space is automatic and predefined, but the user can also define her own spaces and change the rules of how to group items to spaces.

Data Structures

The following data structures are kept:

entry:

- id: serves as key
- timestamp: when entry was created
- user: owner of the entry (i.e., a citizen)
- app: application that created that entry (e.g., runtastic, health-o-matic, ...)
- context: who created this entry (e.g., doctor or any other information, depends on app)
- payload: the actual data of the entry

An entry is any piece of data entered into HealthBank. Example entries are ECGs, results of blood tests, intake of medication, statistics of running in the morning, what I ate for lunch, etc. An entry can be created in many different ways; e.g., entered manually through the web interface, E-Mail, bulk load from an existing database, created by an app on a mobile phone, created by a gadget (e.g., fit bit). The app that generated the entry provides meta-data on how to interpret the payload of the entry; e.g., interpret the payload as a series of GPS coordinates that should be displayed on a map.

user:

- login: name and password
- role: citizen, doctor, etc.

Management of users of the system.

circle:

- owner: owner of the circle (references user)
- circle: e.g., "family" or "doctor" or user-defined
- member: user who the owner accepts to be part of the circle

share:

- entry: references entry
- circle: name of the circle

Users control how to share data with circles. They maintain circles by adding other users to them. Furthermore, they assign entries to circles. Some circles are pre-defined; i.e., "science" and "commercial". If a user specifies that an entry is shared with "science", then the user agrees that the entry may be used by scientist (without actually defining who a scientist is). Likewise, the "commercial" circle is defined by HealthBank.

space:

- name

visualize:

- space: name of space
- entry: reference to entry

Users group entries into spaces. These spaces serve as a way to visualize the information or perform other applications for the user. There are a number of spaces pre-defined; e.g., "all" (gives the whole feed) or "wellness" (contains all the fitness and wellness information). Typically, the "app" of the entry determines the target of an entry, but the user can override such default settings if desired.

Implementation

We envision that "items" are stored in a MongoDB database. Storage of "items" must be scalable (we envision millions of items) and they are poorly structured. All other data structures can be stored in a relational database because they are smaller and highly structure, but may also be stored in MongoDB if that is simpler.

Attachments (0) Show/Add Attachments
Children (0) Hide Children | Add Child Page
Comments (0) Hide Comments | Add comment