

FMK Data Model for Trifork

Gonalo Tomas, Joao Leitao, Nuno Preguia
NOVA-LINCS

30 December 2015

Abstract

In this document we propose an initial version of the data model for the FMK system that considers the specifications in the SyncFree documents.

1 Data Model

Taking the requirements and specifications into consideration, we present our initial version of the data model for the FMK system:

1.1 Patient

- Personal information:
 - Person ID: String
 - Full name: String
 - Address: String
- Treatments: $Set\{t : t \equiv Treatment_{Patient}\}$.
- Prescriptions: $Set\{p : p \equiv Prescription_{Patient}\}$
- Events: $Set\{e : e \equiv Event_{Patient}\}$

Each record will be defined by some personal details like full name, and will hold a set of treatment, prescription and event records that are associated with the patient.

1.2 Treatment

- Treatment ID: String
- Patient name: String
- Creator: MedicalStaff
- Treatment Facility: TreatmentFacility
- Description: String
- Prescriptions: $Set \{p : p \equiv Prescription_{Treatment}\}$.
- Events: $Set \{e : e \equiv Event_{Treatment}\}$.
- Begin date: timestamp
- End date: timestamp

Each treatment has to register its creator, and may store several prescriptions and real life events that happened since the patient started the treatment. Once the patient ends the treatment the end date is set, meaning that some time later it can be deleted.

1.3 Prescription

- Prescription ID: String
- Patient name: String
- Prescriber: MedicalStaff
- Date prescribed: timestamp
- Date processed: timestamp
- Drugs/Elements: $Set \langle String \rangle$
- isDeleted: Bool

Prescriptions must reference the prescribing physician and the patient. It is possible that a prescription is associated with a pharmacy and/or treatment. Prescriptions can be marked as deleted but must never leave the system.

1.4 Event

- Event ID: String
- Patient: Patient
- Treatment: Treatment
- Prescription: String
- Creator: MedicalStaff
- Time of occurrence: timestamp
- Description: String

Events model real life occurrences like taking medication, patient response to drugs and overall clinical progress.

1.5 MedicalStaff

- Practitioner ID: String
- Full name: String
- Address: String
- Specialisation: String
- Treatment facility: TreatmentFacility

MedicalStaff is used to represent authorized medical personnel that can create and update treatments and prescriptions, such as nurses and doctors.

1.6 TreatmentFacility

- Facility ID: String
- Type: String
- Name: String
- Address: String
- Prescriptions: $Set \{p : p \equiv Prescription_{TreatmentFacility}\}$

Treatment facilities represent hospitals and care centers.

1.7 Pharmacy

- Pharmacy ID: String
- Name: String
- Address: String
- Prescriptions: $Set \{p : p \equiv Prescription_{\text{Pharmacy}}\}$

2 System Operations

2.1 Notation

Across this section we use the following semantics:

- The set of all patients is represented by *Patients*.
- The set of all events is represented by *Events*.
- The set of all data-centers is represented as *DataCenters*.
- The set of all pharmacies is represented as *Pharmacies*.
- The set of all treatment facilities is represented as *TreatmentFacilities*.
- The set of all patients as seen by a data-center *DC* is represented by $Patients_{DC}$
- The set of all authorized medical personnel as seen by a data-center *DC* is represented by $MedicalStaff_{DC}$
- The set of all events for a patient *p* is represented by $Events_p$
- The set of all treatments for a patient *p* is represented by $Treatments_p$
- The set of all prescriptions for a patient *p* is represented by $Prescriptions_p$

2.2 SysAdmin Operations

In this subsection we describe some of the operations that are not relative to the system operations themselves, but are otherwise necessary in the system.

2.2.1 Create Patient

This operation adds a patient *patient* to the set of all patients in the context of a data-center *DC*:

Pre-conditions:

$$patient \notin Patients_{DC}$$

Effect:

$$Patients_{DC} = Patients_{DC} \cup patient$$

2.2.2 Create Medical Staff Member

This operation adds an authorized physician or nurse *member* to the system in the context of a data-center *DC*:

Pre-conditions:

$$member \notin MedicalStaff_{DC}$$

Effect:

$$Patients_{DC} = Patients_{DC} \cup member$$

2.2.3 Create Treatment Facility

This operation creates a new TreatmentFacility instance, *facility*:

Pre-conditions:

$$facility \notin TreatmentFacilities$$

Effect:

$$TreatmentFacilities = TreatmentFacilities \cup facility$$

2.2.4 Create Pharmacy

This operation creates a new Pharmacy instance, *pharmacy*:

Pre-conditions:

$$pharmacy \notin Pharmacies$$

Effect:

$$Pharmacies = Pharmacies \cup pharmacy$$

2.3 System Specific Operations

In this subsection we define operations that are relevant for the both the end users and client systems.

2.3.1 Create Treatment

This operation creates a treatment *treatment* for a patient *p* in the context of a data-center *DC*:

Pre-conditions:

$$p \in Patients_{DC}$$

$$treatment \notin Treatments_p$$

Effect:

$$Treatments_p = Treatments_p \cup treatment$$

2.3.2 End Treatment

This operation notifies the system that patient p has finished a treatment t in the context of a data-center DC :

Pre-conditions:

$$p \in Patients_{DC}$$

$$t \in Treatments_p$$

$$t.EndDate = null$$

Effect:

$$t.EndDate = system.getCurrentTime$$

2.3.3 Create Prescription

This operation creates a new prescription $prescription$ for a patient $patient$ in the context of a data-center DC :

Pre-conditions:

$$patient \in Patients_{DC}$$

$$prescription \notin Prescriptions_{patient}$$

Effect:

$$Prescriptions_{patient} = Prescriptions_{patient} \cup prescription$$

2.3.4 Update Prescription

This operation updates a prescription $prescription$ of a patient $patient$ with a new value $prescription'$ in the context of a data-center DC :

Pre-conditions:

$$patient \in Patients_{DC}$$

$$prescription \in Prescriptions_{patient}$$

Effect:

$$prescription = prescription'$$

2.3.5 Add Event

This operation creates a new event e for a patient p in the context of a data-center DC :

Pre-conditions:

$$p \in Patients_{DC}$$

$$e \notin Events_p$$

Effect:

$$Events_p = Events_p \cup e$$