Assignment 5

Course- CS 548
Fall 2016
Submitted By- Yugaank Arun Sharma
Stevens Id- 10419077
Canvas Id- ysharma

Provide a domain-driven design for a clinical information system. You will use the data model you designed in the previous assignment. You should add domain specific logic to the Java classes of the last assignment. Follow the principles of domain-driven architecture in designing your domain model. You are provided with several Maven projects that you should complete for the assignment:

• **ClinicApp:** The umbrella enterprise application project.
• **ClinicDomain**: The domain model for the app.
• **ClinicInit**: Initializes the app during deployment.
• **ClinicRoot**: A Maven parent project for the modules above.

Some of the classes and interfaces that were created and are essential

(1) ClinicDataModel-: Created in previous assignment but more functions, factory aggregates and repositories and several class files are constructed for this assignment like, ITreatmentExporter is created to export treatment records, several functions were defined in different classes like deleteAllPatients or deleteAllProviders etc.
(2) ClinicInt: EJB class created along with a session bean called InitBean. It is Singleton Bean with no interface. InitBean is set up to start when the app is deployed. It is initialized with the app deployment and that is done using @Startup Annotation. When this singleton bean is initialized, several functions are called and their result is displayed on server logs of glassfish. Since we have not set any interface for resources, the logger.info(msg) is used to display the msg if the functions have worked fine.

Domain Logic in the model-:

(i) Add a new patient to the clinic-
The patient entity is created with a factory method. The inputs for this operation should be the patient name, date of birth, age and optionally patient identifier. There is also a check on age, if the age doesn't match with the age calculated by the computer using the date of birth, then an **exception** is caught. The primary key for patient entities will be a separate auto-generated field. The patient DAO then persists the patient entity to the database. If a patient identifier is provided, and a patient with that identifier

already exists in the system, then an exception should be raised and the addition aborted without adding that patient again to the database. If successful, the DAO operation should return the primary key for the new patient object in the database.

(ii) Retrieve a patient aggregate from the system, given the primary key for the Patient-
This function does what its name says, getPatientById, it returns the patient with the id specified. Here Id is the primary key of the patient relation there is no Patient with that id, an exception is caught.

(iii) Retrieve a patient aggregate from the system, given the patient identifier for the Patient-
This function does what its name says, getPatientByPatientId, it returns the patient with the id specified. If there is no Patient with that id, an exception is caught.

(iv) Add a provider to a clinic
A Provider is added to the database with a provider id. If the Id is already defined for the clinic, an exception is raised.

(v) Retrieve a provider aggregate from the clinic-
Retrieves a provider with the provider id or NPI as said by assignment specification, or the normal id of provider.

(vi) Add a treatment for a patient-
The logic used behind this is same as specified in the assignment specification with patient operation sets forward and backward pointers between the patient and treatment entities, while the provider operation sets forward and backward pointers between provider and treatment entities.

(vii) A patient aggregate provides access to its treatment entities-
This logic is done using two operations but also keeping in mid by not violating the encapsulation of the aggregate pattern.

(viii) the provider aggregate should provide these operations for accessing treatments-
Same way, 2 operations but not returning providers to clients directly.

In this submission zip archive-
In the root folder, there is README.pdf, a folder,ClinicApp.ear and video file that demonstrates the deployment of the app.
The folder is named after my first name ‘yugaank’ and inside that folder, the eclipse project is stored.