Microservice Specification (Technical)

'Visit Management'

Visit Management Microservice – Technical Specification based on Microservice Event Sourcing Architecture

REST API Specification

getAllVisitsByDate

Method Type– Get
Request parameter - Date date
response – Encounter(Visit) Entity

---

getAllActiveVisits

Method Type– Get
Request parameter - No Parameter
response – List of Encounter Entity

---

getAllDeactivedVisits

Method Type– Get
Request parameter - No Paramater
response – List of Encounter Entity

---

getAllPatientVisits

Method – Get
Request parameter - No parameter
Response – List of Encounter Entity

---

getPatientVisitsByHealthIdentificationNumber

Method – Get
Request parameter - Patient ID
Response – List of Encounter Entity

---

activatePatientVisitByUniqueHealthIdentificationNumber

Method – Get
Request parameter - uniqueHealthIdentificationNumber

Response – HTTP Status 201(created) , EncounterID

---

deactivatePatientVisitByUniqueHealthIdentificationNumber

Method – Get

Request parameter - uniqueHealthIdentificationNumber

Response – HTTP Status 200(OK) , Encounter ID

createPatientEncounter

Method – POST

Request parameter

HealthUniqueIdentificationNumber
appointmentId
patientArrivalDate
patientArrivalTime
reasonForVisit
episodeId
encounterType

Response – HTTP Status 200(OK) , Encounter ID

updatePatientEncounter

Method – PUT

Request parameter
encounterId
patientArrivalDate
patientArrivalTime
reasonForVisit

Response – HTTP Status 200(OK) , Encounter ID

deactivatePatientEncounter

Method – POST

Request Body
EncounterId
encounterStatusCode

Response Body
HTTP Status 200(OK) , Encounter ID

activatePatientEncounter

Method – POST

Request Body
  EncounterId
encounterStatusCode

Response Body
HTTP Status 200(OK) , Encounter ID

createPatientEpisode

Method – POST

Request parameter

appointmentId
reasonForVisit
HealthIdentificationNumber

Response – HTTP Status 200(OK) , Episode ID

updatePatientEpisode

Method – PUT

Request parameter
episodeId
reasonForVisit

Response – HTTP Status 200(OK) , Episode ID

deactivatePatientEpisode

Method – POST

Request Body

EpisodeId
episodeStatusCode

Response Body
HTTP Status 200(OK) , EpisodeID

activatePatientEpisode

Method – POST

Request Body

episodeId
encounterStatusCode

Response Body

HTTP Status 200(OK) , episode ID

getPatientCohortByDiseaseCode

Method –GET

Request Body

diagnosisCode (ICD10 Code)

Response Body

HTTP Status 200(OK) , List collection of Episode Entity.

Episode Entity (attributes)

Encounter Entity (attributes)

Commands

Visit management service will be capable of handling nine different types of commands

1. **CreatePatientEncounterCommand**

   parameters -

   HealthUniqueIdentificationNumber
   appointmentId
   patientArrivalDate
   patientArrivalTime
   reasonForVisit
   episodeId
   encounterType

2. **UpdatePatientEncounterCommand**

   parameters -

   encounterId
   patientArrivalDate

patientArrivalTime
reasonForVisit

3. **ActivatePatientEncounterCommand**

   parameters -

   EncounterId
   encounterStatusCode

4. **DeactivatePatientEncounterCommand**

   parameters -

   EncounterId
   encounterStatusCode

5. **PatientEncounterVisitCancellationCommand**

   parameters -

   EncounterId
   cancellationReasonCode

6. **CreateEpisodeCommand**

   parameters -

   appointmentId
   reasonForVisit
   HealthIdentificationNumber

7. **ActivateEpisodeCommand**

   parameters -

   episodeId
   episodeStatus

8. **UpdateEpisodeCommand**

   parameters -

   reasonForVisit
   episodeId
   episodeStatus

9. **DeactivateEpisodeCommand**

   parameters -

   episodeId
   episodeStatus

Events

The Visit service will be capable of handling Eight different types of Events-

   1. **PatientEncounterCreatedEvent**

parameters -

HealthUniqueIdentificationNumber
appointmentId
patientArrivalDate
patientArrivalTime
reasonForVisit
episodeId
encounterType

2. PatientEncounterUpdatedEvent

parameters -

encounterId
patientArrivalDate
patientArrivalTime
reasonForVisit

3. PatientEncounterActivatedEvent

parameters -

encounterId
encounterStatus

4. PatientEncounterDeactivatedCommand

parameters -

encounterId
encounterStatus

5. VisitCancellationEvent

parameters -

EncounterId
cancellationReasonCode

6. EpisodeCreatedEvent
parameters -

appointmentId
reasonForVisit
HealthIdentificationNumber

7. EpisodeUpdatedEvent

parameters -

reasonForVisit
episodeId
episodeStatus

8. EpisodeDeactivatedEvent

parameters -

episodeId
episodeStatus

Queries

The Visit service will be capable of handling seven different types of Queries:

1. **getPatientEncounterByuniqueHealthIdentificationNumber**

   Request parameter-   uniqueHealthIdentificationNumber
   Response – Encounter Value Object

2. **getAllPatientActiveEncounters**

   Request parameter-   Nil
   Response – List collection of Encounter Value Objects

3. **getAllPatientDeactivatedEncounters**

   Request parameter-   Nil

   Response – List collection of Encounter Value Objects

4. **getPatientEpisodeByuniqueHealthIdentificationNumber**

   Request parameter-   uniqueHealthIdentificationNumber
   Response – Episode Value Object

5. **getAllPatientActiveEpisodes**

   Request parameter-   Nil
   Response – List collection of Episode Value Objects

6. **getAllPatientDeactivatedEpisodes**

   Request parameter-   Nil

   Response – List collection of Episode Value Objects

7. **getPatientCohortGroupsByDiseaseCode**

   Request parameter - Unique Health Identifier, diagnosisCode (ICD 10 Code)

   Response parameter – List of Episode Value Objects