Patient

- patientId : int
- patientRecords : List<PatientRecord>

+ Patient(int patientId)
+ addRecord(double measurementValue, String recordType, long timestamp): void
+ getRecords(long startTime, endTime) : List<PatientRecord>

AlertGenerator

-dataStorage : DataStorage

+ AlertGenerator(DataStorage dataStorage)
+ evaluateData(Patient patient): void
- triggerAlert(Alert alert) : void

Alert

- patientId: String
- condition : String
- timestamp : long

+ Alert(String patientId, condition, long timestamp)
+ getPatientId() : String
+ getCondition() : String
+ getTimeStamp() : long

AlertManager

- staffRecipient : List<Staff>
- alertQueue : Queue<Alert>

+ alertManager()
+ dispatchAlert(Alert alert) : void
+ dispatchAllAlerts() : void

DataStorage

- patientMap: Map<Integer, Patient>

+ DataStorage()
+addPatientData(int patientId, double measurementValue, String recordType, long timestamp): void
+ getRecords(int patientId, long startTime, long endTime) : List<PatientRecord>
+ getAllPatients : List<Patient>
+ main(String[] args) : void

Staff

- nameId : int
- departmentNum : int

+ getNameId() : int
+ getDepartmentNum() : int

Use

Use

Use

association

association

association

1

0..1

0..1

0..*

0..*

0..*

1..*

0..*

1..*

The patient class represents individual patients and stores their medical records, each patient is identified by an Id. The list allows access to old data efficiently. The method getRecord provides access to records withing a time range. This setup ensures modularity, by using patient data and operations within the same class. The Alert class is a container for alert informations. The AlertGenerator class evaluates patient data using attributes stored in DataStorage, analyzes patient record and gives alerts if necessary. The AlertManager class is used to send alerts to staff using a queue it supports individual and wholly processing of alerts. The DataStorage class uses a map to efficiently patient finding, the association with Patient and Alert is enables easy navigation between data.