Software Lab
Assignment-3

Pratyush Anand
2010EEY7515
C.Tech, IITD

January 24, 2012

Chapter 1

Problem Statement

Generate UML diagrams for following scenarios

- Use case Diagram
- Sequence Diagram
- Class Diagram

Chapter 2

Operation

2.0.1 Assumptions

- Middleware software also contains family patient portal
- Accounting software has been considered as an external agent.

2.0.2 Use case Diagram

- All the external part of the system has been represented as actors and has been kept outside the rectangular boundary, while all use cases has been kept inside the rectangular boundary. So, following actors has been identified.

  Doctor : Can view notifications on hid PDA and visits patient's room
  Views Patient's report and updates it.
  Orders test if required.
  Can view appointment request.
  Can grant appointment to relative.

  Nurse:Can view Notification
  Can see triage list and prioritize service.
  can see doctor's Location

  Accounting Application: Can bill doctors visit
  Can bill ordered test.

  Relatives: Can see doctor's presence.
  Can request doctor's appointment.
  Can connect Doctor.

  Sensors: Can monitor patient and can generate and alarm.

  Patient: Can see doctors presence and can raise alarm

Other than the above ¡¡extend¿¿ use cases some has been taken as ¡¡uses¿¿ use cases.
Determining Location and issue interrupt: This will be used by most of the use cases as shown in use case diagram.
Similarly other ¡¡uses¿¿ use cases are also used by other use cases.

2.0.3 Class Diagram

- objects defined in sequence diagram has been represented here with its attributes and methods.
- Here, it is clearly visible that middleware software is the central agency with whom every one interacts. For example, doctor, nurse and patient all are updating their location continuously.So they will just call send_location and update_location_for_all will update the location. Now this updated location can be viewed by either calling view_doctors_presence or view_patients_location.
- Class Doctor: It has following attributes:
  Location: It will have information about his location.
  Name: Will have doctor's name.
  ID: Will have employee ID.
  This class has following operations.
  view_report: Can view report of the patient.
  update_report: It can also invalidate and update report.
  order_test: It can order test if required.
  view_appointment: Can view appointment requested by relatives. grant_appointment: Can grant appointment to relative.
  view_emergency_notifications: View alerts generated by either patient or sensor.
  send_location: will always sends its location.
- Class Nurse: It has following attributes:
  Location: It will have information about her location.
  Name: Will have nurse's name.
  ID: Will have employee ID.
  This class has following operations.
  view_emergency_notifications: View alerts generated by either patient or sensor.
  send_location: will always sends its location.
  view_triage_list:Can see all the list.
  prioritize_service: Can prioritize on need basis.
  view_doctors_location: Can see doctor's location by passing his IDs.
  thank_others: Can thank other nurse by passing her ID.
- Class Patient: It has following attributes:
  Location: It will have information about her location.
  Name: Will have his name.
  ID: Will have patient ID.
  This class has following operations.
  view_doctors_presence: Can view if doctor is making a round. generate_alarm: Can generate alarm in case of emergency. send_location: will

always sends its location.

- Class Accounting Application:This class has following operations.
  bill_doctors_visit: bill for each visit of doctor. bill_ordered_test: Bill for each test ordered by doctor.

- Class relative: It has following operations.
  view_doctors_presence: Can view if doctor is making a round.
  request_appointment: Can request appointment from doctor.
  connect_doctor: Can connect to doctor if appointment granted.

- Class sensors: It has following operations.
  monitor_patient: can monitor patient's health.
  generate_alarm: Can generate alarm in case of emergency.

- Class Middleware Software: It has following operations.
  generate_report: Will sense doctor's presence in patient's room and will generate patient's report for the doctor.
  update_test_request: Will update test request if it is done by doctor, and will send updated request to accounting system.
  update_request_appointment: sends appointment information to doctor.
  update_granted_appointment: Sends granted appointment information to relatives.
  update_location_for_all: update location database and send interrupt to doctor.
  receive_and_send_private_messages: Receive and send messages like thanks.
  receive_and_generate_emergency_requests:
  update_doctors_visit: send doctors visit to accounting systems.

2.0.4 Sequence Diagram

- All the actors of use cases along with middleware software has been taken as objects.
- All objects asks for required information from middleware software. They also send the updated information to middleware software.