COEN 275 – Spring 2017

Object Oriented Analysis, Design and Programming

So Safe Home Security System

Deliverable- 2

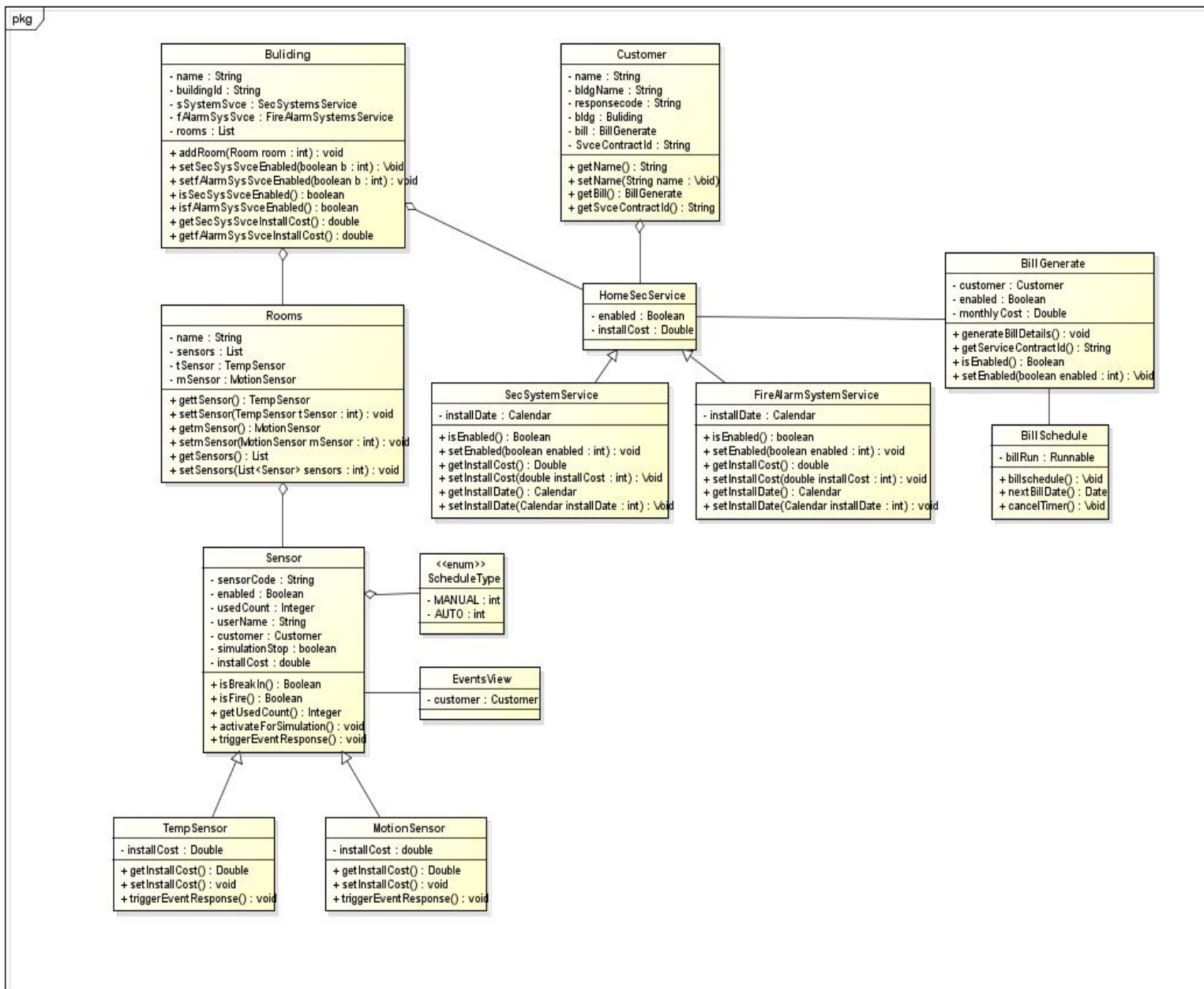(Class Diagram, Sequence Diagrams, Logical Partitioning)

Submitted By

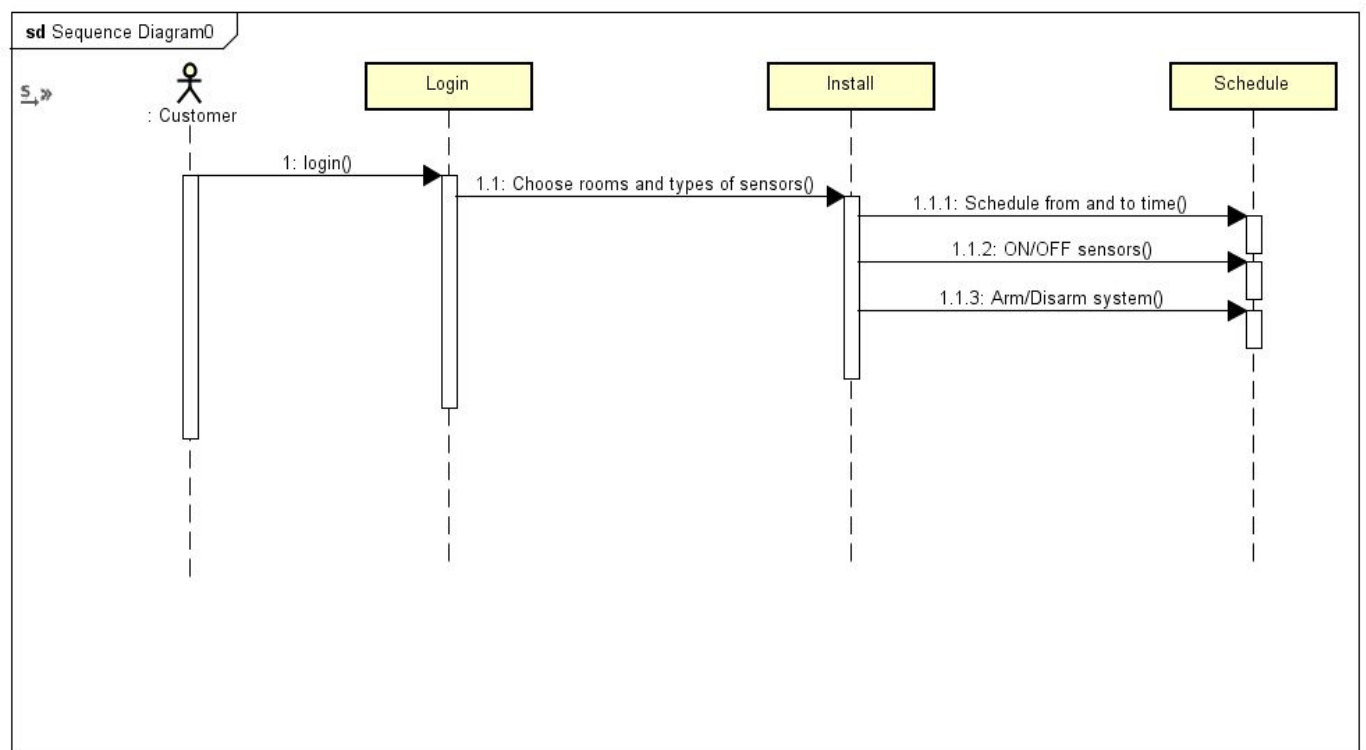Haritha Mannuru
Ramya Sivaraman

Table of contents:

1. Class Diagram

pkg

Buliding

- name : String
- buildingId : String
- sSystemSvce : SecSystemsService
- fAlarmSysSvce : FireAlarmSystemsService
- rooms : List

+ addRoom(Room room : int) : void
+ setSecSysSvceEnabled(boolean b : int) : Void
+ setfAlarmSysSvceEnabled(boolean b : int) : void
+ isSecSysSvceEnabled() : boolean
+ isfAlarmSysSvceEnabled() : boolean
+ getSecSysSvceInstallCost() : double
+ getfAlarmSysSvceInstallCost() : double

Customer

- name : String
- bldgName : String
- responsecode : String
- bldg : Buliding
- bill : BillGenerate
- SvceContractId : String

+ getName() : String
+ setName(String name : Void)
+ getBill() : BillGenerate
+ getSvceContractId() : String

Bill Generate

- customer : Customer
- enabled : Boolean
- monthlyCost : Double

+ generateBillDetails() : void
+ getServiceContractId() : String
+ isEnabled() : Boolean
+ setEnabled(boolean enabled : int) : Void

HomeSecService

- enabled : Boolean
- installCost : Double

Rooms

- name : String
- sensors : List
- tSensor : TempSensor
- mSensor : MotionSensor

+ gettSensor() : TempSensor
+ settSensor(TempSensor tSensor : int) : void
+ getmSensor() : MotionSensor
+ setmSensor(MotionSensor mSensor : int) : void
+ getSensors() : List
+ setSensors(List<Sensor> sensors : int) : void

SecSystemService

- installDate : Calendar

+ isEnabled() : Boolean
+ setEnabled(boolean enabled : int) : void
+ getInstallCost() : Double
+ setInstallCost(double installCost : int) : Void
+ getInstallDate() : Calendar
+ setInstallDate(Calendar installDate : int) : Void

FireAlarmSystemService

- installDate : Calendar

+ isEnabled() : boolean
+ setEnabled(boolean enabled : int) : void
+ getInstallCost() : double
+ setInstallCost(double installCost : int) : void
+ getInstallDate() : Calendar
+ setInstallDate(Calendar installDate : int) : void

Bill Schedule

- billRun : Runnable

+ billschedule() : Void
+ nextBillDate() : Date
+ cancelTimer() : Void

Sensor

- sensorCode : String
- enabled : Boolean
- usedCount : Integer
- userName : String
- customer : Customer
- simulationStop : boolean
- installCost : double

+ isBreakIn() : Boolean
+ isFire() : Boolean
+ getUsedCount() : Integer
+ activateForSimulation() : void
+ triggerEventResponse() : void

<<enum>>
ScheduleType

- MANUAL : int
- AUTO : int

EventsView

- customer : Customer

TempSensor

- installCost : Double

+ getInstallCost() : Double
+ setInstallCost() : void
+ triggerEventResponse() : void

MotionSensor

- installCost : double

+ getInstallCost() : Double
+ setInstallCost() : void
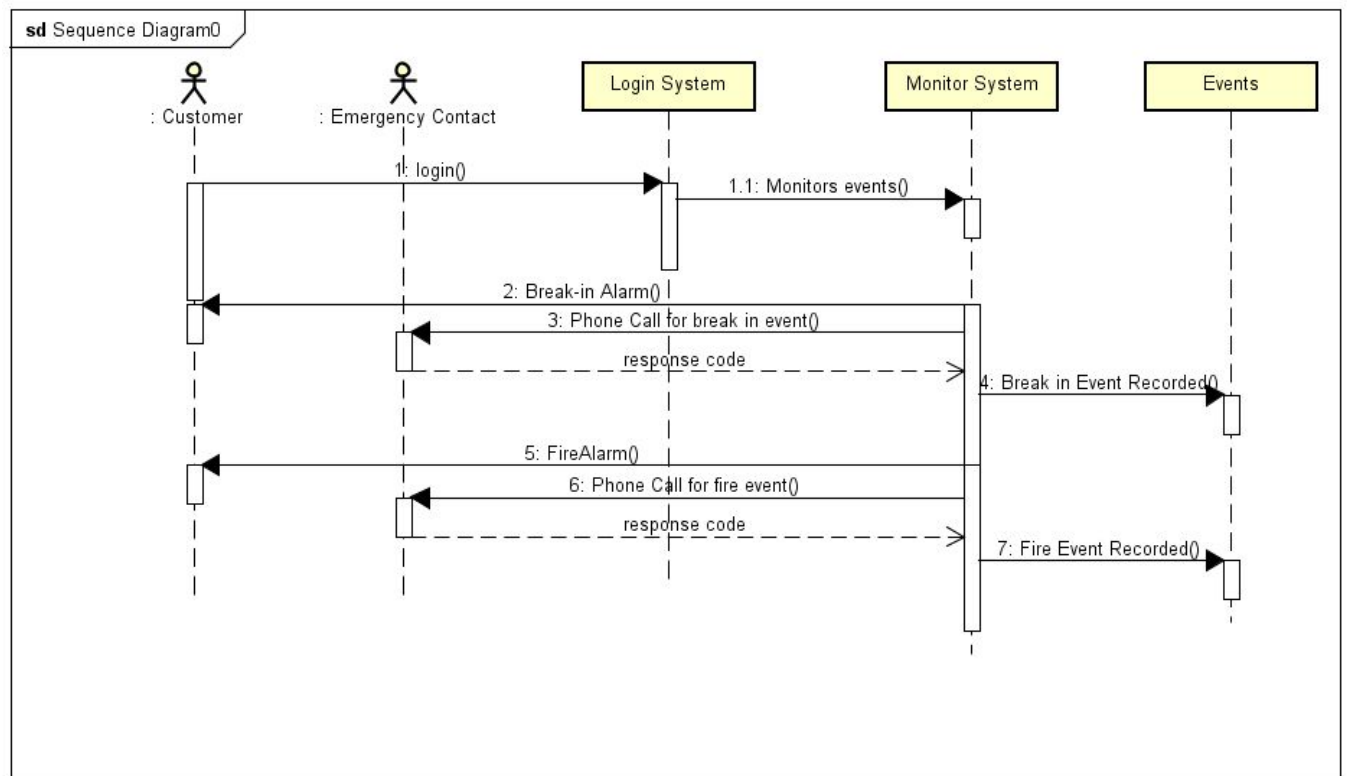+ triggerEventResponse() : void
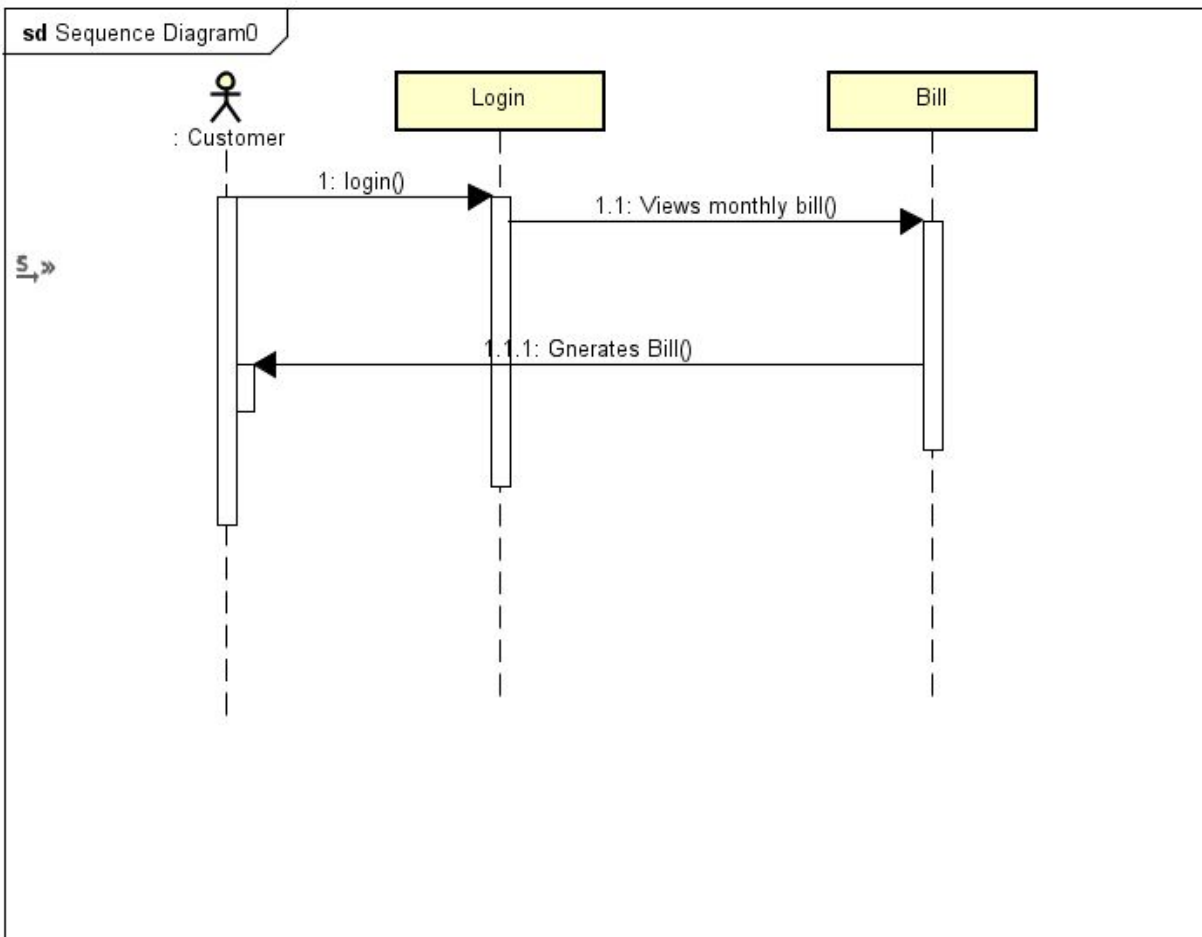
2. Sequence Diagrams

2.1 Sequence diagram for Use Case Login, Install and Schedule Sensors

2.2 Sequence diagram for Use Case Monitor and Simulation

sd Sequence Diagram0

		Login System	Monitor System	Events

: Customer

: Emergency Contact

1: login()

1.1: Monitors events()

2: Break-in Alarm()

3: Phone Call for break in event()

response code

4: Break in Event Recorded()

5: FireAlarm()

6: Phone Call for fire event()

response code

7: Fire Event Recorded()

2.3 Sequence diagram for Use Case Billing

sd Sequence Diagram0

: Customer   Login   Bill

1: login()

1.1: Views monthly bill()

1.1.1: Gnerates Bill()

3. Logical Partitioning of system into Packages

LOGICAL LAYER