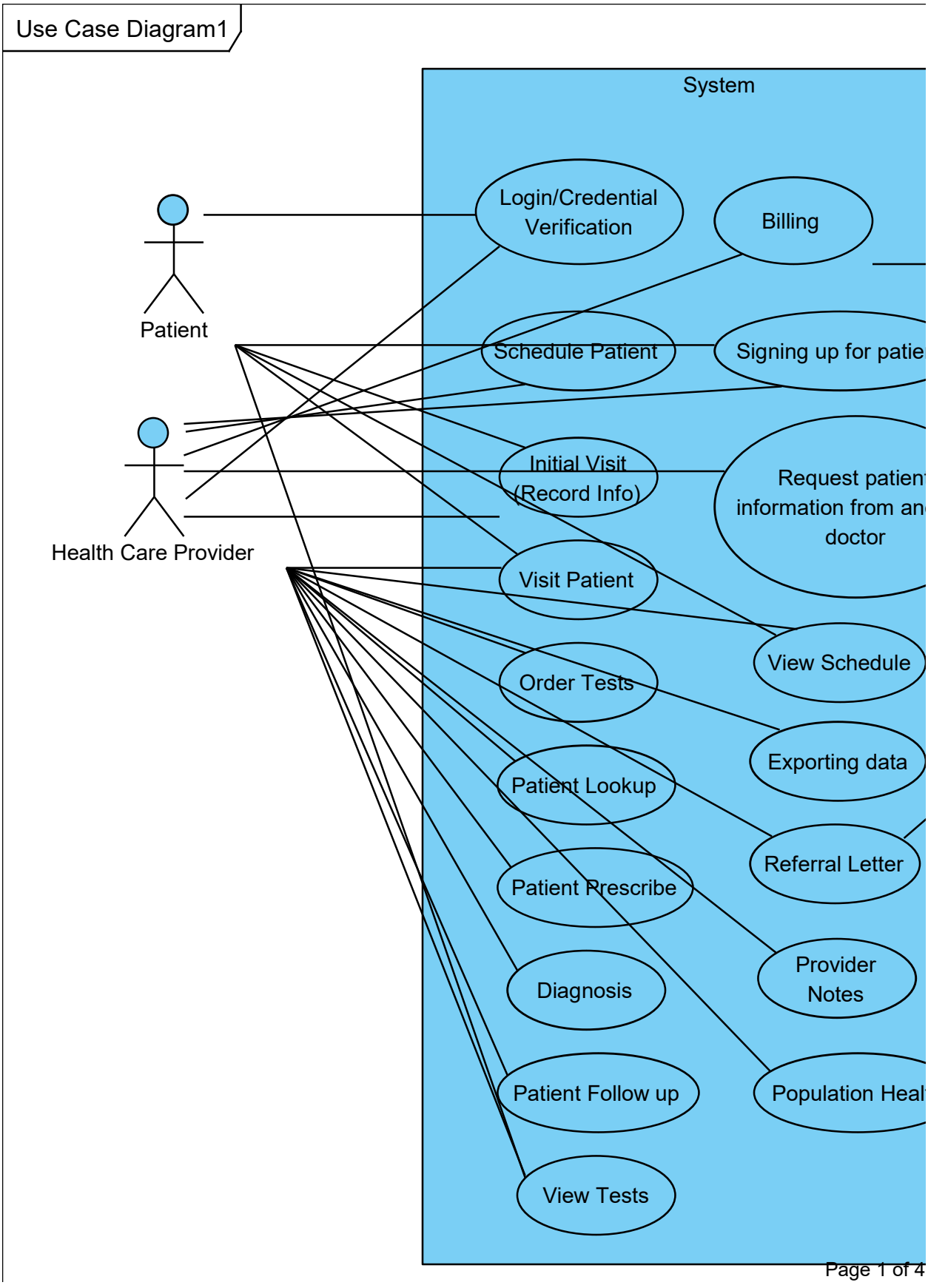
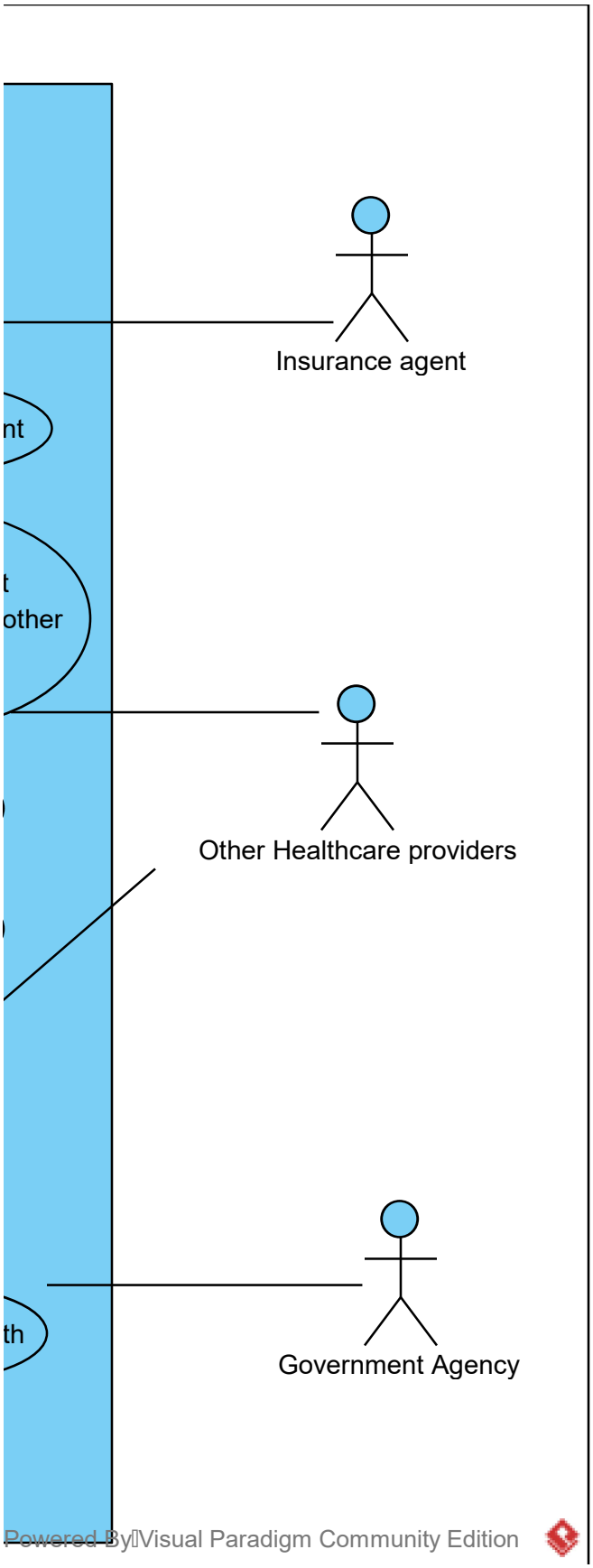

Use Case Diagram1
System
Patient
Health Care Provider
Login/Credential Verification
Billing
Schedule Patient
Signing up for patie
Initial Visit (Record Info)
Request patien information from an doctor
Visit Patient
View Schedule
Order Tests
Exporting data
Patient Lookup
Referral Letter
Patient Prescribe
Provider Notes
Diagnosis
Patient Follow up
Population Heal
View Tests



Billing

Information

Rank	Unspecified
ID	
Status	Unspecified
Justification	
Primary Actors	Health Care Provider, Insurance agent
Suppporting Actors	

Scenarios

Scenario

1. Total cost for service rendered is calculated
2. Services rendered are included on bill with cost
3. Day by which bill must be payed is included
4. Bill is sent to necessary parties (i.e. insurance, government taxing, patient)
5. System records payments made

Extensions

1.a. Invalid values are entered
  1. System signals error and rejects entry
  2. Office accountant responses to error
    2.1. Data for charging is saved in human readable format
      2.1.1. Office accountant manually enters data
      2.1.2. System recalculates and displays results
    2.2. Necessary data is not saved
      2.2.1. Accountant reports error
      2.2.2. Operation is put on hold until correct data is recovered

2.a. Services rendered are not available
  1. System reports error
  2. Accountant responds to errors
    2.1. Services are locatable
      2.1.1. Account enters data by hand
    2.2. Services are not locatable
      2.2.1. Accountant reports error
      2.2.2. Operation halted until correct data is recovered

3.a. Patient receives payment extension for extenuating circumstances

3.a. Patient receives payment extension for extenuating circumstances
   1. The extension is noted on the bill
   2. The extension is saved in the system and reported to other necessary services (insurance, government taxing)

4.a. Bill is unable to be sent due to extenuating system errors
   1. System reports error
   2. Necessary attendant (i.e. office accountant) manually sends data

Details

Level	User goal
Complexity	N/A
Use Case Status	N/A
Implementation Status	N/A
Preconditions	Patient provided a service.
Post-conditions	N/A
Author	N/A
Assumptions	N/A

Requirements

Different fields for different types of patient visits

Doctor Database

insurance code

Look up

Access

Data consistency

Easy file sharing

Diagnosis

Information

Rank	Unspecified
ID	
Status	Unspecified
Justification	
Primary Actors	Health Care Provider
Suppporting Actors	

Scenarios

Scenario

1. Doctor diagnoses patient
2. Doctor opens patient's file
3. System creates new diagnosis for patient
4. Doctor enters patient's diagnosis
5. System saves and stores diagnosis in patient folder and in other necessary files

Extensions

2.a. Patients file is unable to open
   1. Doctor restarts system
   2. System reports error and continues in clean state

2.b. Doctor does not have permission to access file

4.a. Patient has additional ailments and requires more than one diagnosis
   1. Doctor creates multiple diagnoses
   2. System records multiple diagnoses and stores data under same visit

5.a. Patient requests copy of diagnosis
   1. System provides available information
   2. Doctor prints or sends the information to the patient in a secure way

#.a. At any time, System fails
   1. Doctor restarts system
   2. System reconstructs former state
      2.1. System detects anomaly
         2.1.1. Error is reported to doctor and IT and enters a clean state
         2.1.2. Doctor creates new diagnosis

Details

Details

Level	user goal
Complexity	N/A
Use Case Status	N/A
Implementation Status	N/A
Preconditions	Visit Patient
Post-conditions	N/A
Author	N/A
Assumptions	N/A

Requirements

Different fields for different types of patient visits

Look up

Access

Input

Data consistency

Easy file sharing

Population health

Initial Visit (Record Info)

Information

Rank	Unspecified
ID	
Status	Unspecified
Justification	
Primary Actors	Patient, Health Care Provider
Suppporting Actors	

Scenarios

Scenario

1. Patient walks into doctor's office for a consultation
2. Patient checks in with staff, identifying themself
3. Staff check to see whether customer health sheet is complete
4. Patient health sheet is complete
5. Staff will check doctor's availability
6. Doctor is available
7. Patient is sent to doctor
8. Patient is evaluated by doctor
9. Doctor does not need to update health sheet

Extensions

2.a. Patient cannot identify themself and is a walk-in
   1. Patient is refused

2.b. Patient cannot identify themself but has an appointment
   1. Patient logs into the system
   2. Patient checks into appointment through the system

4.a. Patient health sheet is empty/incomplete
   1. Patient logs into system
   2. Patient fills any empty/incomplete sections of health sheet

6.a. Doctor is unavailable
   1. Staff gives patient estimate as to when doctor will be available
   2. Staff asks patient to wait in waiting room

9.a. Doctor needs to update health sheet
   1. Doctor gives staff the information necessary to update the health sheet
   2. Staff logs into the system

2. Staff logs into the system
3. Staff makes necessary changes to the health sheet

#.a. anytime system does not respond

1. Staff will restart system

Details

Level	User goals
Complexity	N/A
Use Case Status	N/A
Implementation Status	N/A
Preconditions	Login/Credential Verification
Post-conditions	Patient information is recorded.
Author	N/A
Assumptions	N/A

Requirements

Doctor-to-Patient

insurance code

Patient - User Interface

Input

Data consistency

Login/Credential Verification

Information

Rank	Unspecified
ID	
Status	Unspecified
Justification	
Primary Actors	Health Care Provider, Patient
Suppporting Actors	

Scenarios

Scenario

1. Patient accesses system
2. System prompts client to enter their credentials
3. Patient enters their credentials
4. System checks e-mail username to see if an account is associated
5. Account is associated
6. System checks password to see if it matches the account password
7. Password is correct
8. System redirects patient to patient home page
9. After doing what is necessary, patient selects to sign out
10. System locks the user out, requiring credentials to sign in again

Extensions

3.a. Patient to does not yet have an account
   1. Patient selects option to create an account
   2. System redirects patient to account creation

5.a. E-mail entered does not match an account
   1. System prompts patient to re-enter credentials
      1.1. Patient re-enters credentials until correct
      1.2. Patient does not have an account
         1.2.1. Patient chooses to create an account

7.a. Password entered does not match account password
   1. System prompts patient to re-enter password
      1.1. Patient re-enters credentials until correct
      1.2. Patient does not have an account
         1.2.1. Patient chooses to create an account

1.2.1. Patient chooses to create an account

Details

Level	User goal
Complexity	N/A
Use Case Status	N/A
Implementation Status	N/A
Preconditions	N/A
Post-conditions	Patient gains access to the system
Author	N/A
Assumptions	N/A

Requirements

Patient - User Interface

Look up

Input

Patient Database

Order Tests

Information

Rank	Unspecified
ID	
Status	Unspecified
Justification	
Primary Actors	Health Care Provider
Suppporting Actors	

Scenarios

Scenario

1. Doctor opens window to order test.
2. Doctor selects patient in need of test.
3. Doctor selects test to perform.
4. Doctor sends order to lab.
5. Lab receives order successfully.

Extensions

2.a. Doctor closes window while selecting patient.
   1. Window closes, patient data remains unchanged
   2. Doctor may reopen window and select patient again

5.a. Lab does not receive test order
   1. Doctor may send order again

Details

Details

Level	user goal
Complexity	N/A
Use Case Status	N/A
Implementation Status	N/A
Preconditions	Login/Credential Verification
Post-conditions	Necessary test for patient is determined. Doctor sends test order. Lab receives order.
Author	N/A
Assumptions	N/A

Requirements

Doctor-to-Patient

Doctor Database

Time sheet

Orders

Interaction with a lab which will receive test orders

Patient Follow up

This use case details the EHR use of following up with a patient after their appointment in order to provide patients with information that they would need.

Information

Rank	Unspecified
ID	
Status	Unspecified
Justification	
Primary Actors	Health Care Provider
Suppporting Actors	

Scenarios

Scenario

1. Doctor concludes visit with patient
2. System prompts doctor to enter follow up information.
3. Doctor enters follow up information as well as other information doctor wishes to share with patient
4. System bundles doctors follow up notes with prescription information and test information
5. System sends message to patient to view their follow up information
6. Patient clicks on follow up panel
7. Patient views doctors notes
8. Patient exits follow up panel

Extensions

3.a. Follow up information has not all been created yet
   1. System notes that this information needs to be collected and is still waiting
   2. System displays blank information for these fields
   3. Once information has been collected, system updates record with proper information

6.a. Follow up panel is not updated
   1. System prints error message

7.a. Doctors notes are not complete
   1. System notes blanks in results and fills in information as it comes in

#.a. At any time user chooses to exit program
   1. System saves information needed and returns to main screen

Details

Level	N/A
Complexity	N/A
Use Case Status	N/A
Implementation Status	N/A
Preconditions	Visit PatientDiagnosisOrder TestsPatient PrescribeInitial Visit (Record Info)Patient Lookup
Post-conditions	N/A
Author	N/A
Assumptions	N/A

Requirements

Doctor-to-Patient

Patient - User Interface

Patient Lookup

Information

Rank	Unspecified
ID	
Status	Unspecified
Justification	
Primary Actors	Health Care Provider
Suppporting Actors	

Scenarios

Scenario

1. Doctor enters a patient name into search bar.
2. Doctor clicks on patient name.
3. Patient information is displayed.
4. Doctor exits window.

Extensions

1.a. Patient has no record in database
   1. System displays no results.
   2. Doctor may search a new name

Details

Level	user goal
Complexity	N/A
Use Case Status	N/A
Implementation Status	N/A
Preconditions	Login/Credential Verification
Post-conditions	N/A
Author	N/A
Assumptions	N/A

Requirements

Requirements

Look up

Patient Database

Simple to use UI for Doctor

Results must display quickly

Patient Prescribe

Information

Rank	Unspecified
ID	
Status	Unspecified
Justification	
Primary Actors	Health Care Provider
Suppporting Actors	

Scenarios

Scenario

1. Doctor determines prescription
2. Electronic prescription is created
3. Doctor stores details of prescription on patients file
4. System stores electronic prescription
5. Prescription is added to patients file
6. System prints out receipt or sends to corresponding pharmacy

Extensions

1.a. Patient does not receive prescription from diagnosis
   1. A prescription is not created

2.a. Electronic prescription does not generate
   1. Doctor closes system and restarts it
   2. Doctor reenters credentials and tries again
      2.1. If system still refuses to work, prior system state is loaded and necessary departments are notified

3.a. Patients file won't open
   1. Doctor closes system and restarts it
   2. If system still doesn't work notify necessary departments
      2.1. Store data on paper until system works correctly

4.a. System is unable to store electronic prescription correctly
   1. System state is saved
   2. Doctor is notified of error
   3. Doctor notifies correct department
      3.1. Data is saved on paper until system can save correctly

5.a. System stores patient data in wrong patient file

5.a. System stores patient data in wrong patient file
1. System alerts doctor and IT of error
2. Data is saved but removed from incorrect file
3. Doctor reenters data or saves it on paper

6.a. Prescription is unable to be shared with pharmacy
1. System lets doctor know error
2. System restarts but saves current state
   2.1. Doctor's office manually calls in prescription if system is down for time being

#.a. Anytime system fails
1. Data is stored on paper to later be input to the system
2. System is restart by IT
3. System reconstructs to prior state
   3.1. If system detects anomalies in system, sends message to doctor and corresponding

Details

Level	user goal
Complexity	N/A
Use Case Status	N/A
Implementation Status	N/A
Preconditions	Patient Prescribe
Post-conditions	N/A
Author	N/A
Assumptions	N/A

Requirements

Requirements

Different fields for different types of patient visits

Clinical Test

Orders

Access

Data consistency

Easy file sharing

Population health

Population Health

This use case calculates population health parameters for reporting to government agencies and for statistical purposes.

Information

Rank	Unspecified
ID	
Status	Unspecified
Justification	
Primary Actors	Health Care Provider, Government Agency
Suppporting Actors	

Scenarios

Scenario

1. User clicks on the population health parameter list
2. System displays a list of population health parameters for which the user can look for.
   - 2.1. Parameters include: Number of patients presenting with any particular condition, dates with a higher than normal number of patient visits, demographics of patients, vital signs, and test results for the population of patients.
3. System calculates results based on the parameters provided.
4. System displays results
   - 4.1. User can choose to adjust display options such as date ranges or graph coordinates.
   - 4.2. System will modify results to accommodate this
5. System allows for users to download/ save a copy of the results

Extensions

- 3.a. System fails to calculate results based on the parameters provided
  1. System prints an error message and returns to population health parameter list
- 4.a. System fails to display results
  1. System prints an error message and returns to the population health parameter
- 5.a. System fails to download correctly
  1. System prints an error.
- #.a. At any time the user can choose to leave this panel
  1. System will exit this panel without any data loss

Details

Level	N/A
Complexity	N/A
Use Case Status	N/A
Implementation Status	N/A
Preconditions	Login/Credential Verification
Post-conditions	N/A
Author	Sam Shenoi
Assumptions	N/A

Requirements

Population health

All data is de-identified in order to protect patient privacy

Provider Notes

Information

Rank	Unspecified
ID	
Status	Unspecified
Justification	
Primary Actors	Health Care Provider
Suppporting Actors	

Request patient information from another doctor

Information

Rank	Unspecified
ID	
Status	Unspecified
Justification	
Primary Actors	Health Care Provider, Other Healthcare providers
Suppporting Actors	

Scenarios

Scenario

1. Request for Doctor's name or id.
2. Enters correct information.
3. Display and ask whether the doctor the user are trying to access is correct.
4. User approves.
5. System retrieves the doctor information.
6. System parses and display readable information to user.

Extensions

2.a. user enters incorrect information
   1. Reply with an error message and to try again.
   2. User enters correct information.
   3. Go to 3 in main scenario.

4.a. user declines
   1. Take user back to the number 1 of the main scenario

5.a. an error occurs retrieving the information
   1. Send error message to technician.
   2. Tell user to try again later.

Details

Details

Level	user goals
Complexity	N/A
Use Case Status	N/A
Implementation Status	N/A
Preconditions	Login/Credential Verification
Post-conditions	N/A
Author	N/A
Assumptions	N/A

Requirements

Different fields for different types of patient visits

Doctor-to-Patient

Doctor Database

Look up

Access

Easy file sharing

Schedule Patient

Information

Rank	Unspecified
ID	
Status	Unspecified
Justification	
Primary Actors	Health Care Provider
Suppporting Actors	

Scenarios

Scenario

1. Patient asks to schedule appointment
2. Staff asks patient for credentials
3. Patient provides correct credentials to staff
4. Staff asks patient which time is convenient for appointment
5. Patient gives staff valid time for appointment
6. Staff enters new appointment into schedule at requested time
7. Schedule updated with new appointment

Extensions

2.a. This is patient's first visit
   1. Proceed with first visit protocol

3.a. Patient provides incorrect credentials
   1. If patient does not have account, patient signs up for one
   2. Patient tries again until correct credentials are provided

5.a. Appointment time given is invalid
   1. Patient is prompted for a new, available time until correct time is provided.

Details

Details

Level	User goal
Complexity	N/A
Use Case Status	N/A
Implementation Status	N/A
Preconditions	N/A
Post-conditions	Patient appointment added to schedule
Author	N/A
Assumptions	N/A

Requirements

Different fields for different types of patient visits

Doctor Database

Time sheet

insurance code

Patient - User Interface

Input

Signing up for patient

Information

Rank	Unspecified
ID	
Status	Unspecified
Justification	
Primary Actors	Health Care Provider, Patient
Suppporting Actors	

Scenarios

Scenario

1. Patient arrives to sign up page.
2. Request patient's full name, email, DOB, sex, height, and weight.
3. Patient enters correct information.
4. Then request for patient contact number, marital status, and address.
5. Patient continues to enter correct information.
6. Then request for patient emergency contact information.
7. Patient enters emergency contact information.
8. Ask whether patient is taking any medication.
9. Patient answers no.
10. Ask whether patient has any current or past health conditions.
11. Patient answer no.
12. Request patient to make password.
13. Patient creates password.
14. Create patient id number.
15. Patient information is pushed onto the patient database.
16. Patient is then fully signed up.
17. Patient is then met with a sign-up completion window that shows their id.

Extensions

3.a. Patient enters invalid email address
   1. The system request patient to enter a correct email
   2. Patient enters correct email

9.a. Patient says yes
   1. Request medication patient is taking
   2. Patient enters medication

2. Patient enters medication

11.a. Patient says yes

1. Request for patients past or current health conditions
2. Patient enters past or current health conditions

Details

Level	User Goal
Complexity	N/A
Use Case Status	N/A
Implementation Status	N/A
Preconditions	N/A
Post-conditions	N/A
Author	N/A
Assumptions	N/A

Requirements

Different fields for different types of patient visits

Patient - User Interface

Access

Input

Data consistency

Easy file sharing

View Schedule

Information

Rank	Unspecified
ID	
Status	Unspecified
Justification	
Primary Actors	Health Care Provider, Patient
Suppporting Actors	

Scenarios

Scenario

1. Ask whether user is a doctor or patient.
2. User enters type.
3. System then prompt user to enter id number and password.
4. User enters correct credentials
5. System fetches schedule information
6. Display schedule information.

Extensions

4.a. User enters wrong credentials
   1. Prompt user to enter credentials again.
   2. User enters correct credentials
   3. Go to 5 in main scenario.

5.a. user is a doctor.
   1. Doctor information is fetched.
   2. Doctor schedule and information is displayed.

5.b. user is a patient
   1. Patient information is fetched.
   2. Patient schedule and information is displayed

Details

Details

Level	user goal
Complexity	N/A
Use Case Status	N/A
Implementation Status	N/A
Preconditions	Login/Credential VerificationSchedule Patient
Post-conditions	N/A
Author	N/A
Assumptions	N/A

Requirements

Different fields for different types of patient visits

Doctor Database

Time sheet

Patient - User Interface

Access

Data consistency

Easy file sharing

View Tests

This use case determines how a patient or a provider can view test results that were previously conducted.

Information

Rank	Unspecified
ID	
Status	Unspecified
Justification	
Primary Actors	Health Care Provider, Patient
Suppporting Actors	

Scenarios

Scenario

1. User clicks on view tests.
2. System displays list of test dates
3. User selects test date
4. System displays all of the test results of all tests associated with that test date
5. User unselects test date
6. System returns to previous state

Extensions

3.a. User selects view test results over time
   1. System displays a graph of each type of test results over time
   2. User decides to minimize the range of test results viewed
      2.1. System redraws graph to accommodate this

#.a. At any time, user decides to exit test result panel
   1. System gracefully leaves panel without doing anything to the data

Details

Details

Level	N/A
Complexity	N/A
Use Case Status	N/A
Implementation Status	N/A
Preconditions	Login/Credential VerificationOrder Tests
Post-conditions	N/A
Author	N/A
Assumptions	N/A

Requirements

Patient - User Interface

Orders

Look up

Data consistency

Visit Patient

Information

Rank	Unspecified
ID	
Status	Unspecified
Justification	
Primary Actors	Patient, Health Care Provider
Suppporting Actors	

Scenarios

Scenario

1. Doctor clicks to open up a window for recording the patient's data and condition.
2. Doctor asks the patient about his/her condition.
3. Doctor records information in the open window.
4. Doctor saves the file.
5. Doctor exits the window.

Extensions

3.a. Doctor exits window while recording data
   1. Doctor will be prompted to save the file before exiting

4.a. File fails to save correctly
   1. Information remains unchanged
   2. Doctor may attempt to save again

Details

Details

Level	user goal
Complexity	N/A
Use Case Status	N/A
Implementation Status	N/A
Preconditions	Initial Visit (Record Info)Schedule PatientLogin/Credential Verification
Post-conditions	Record of visit is saved. Doctor is aware of the patient’s condition and can diagnose accurately.
Author	N/A
Assumptions	N/A

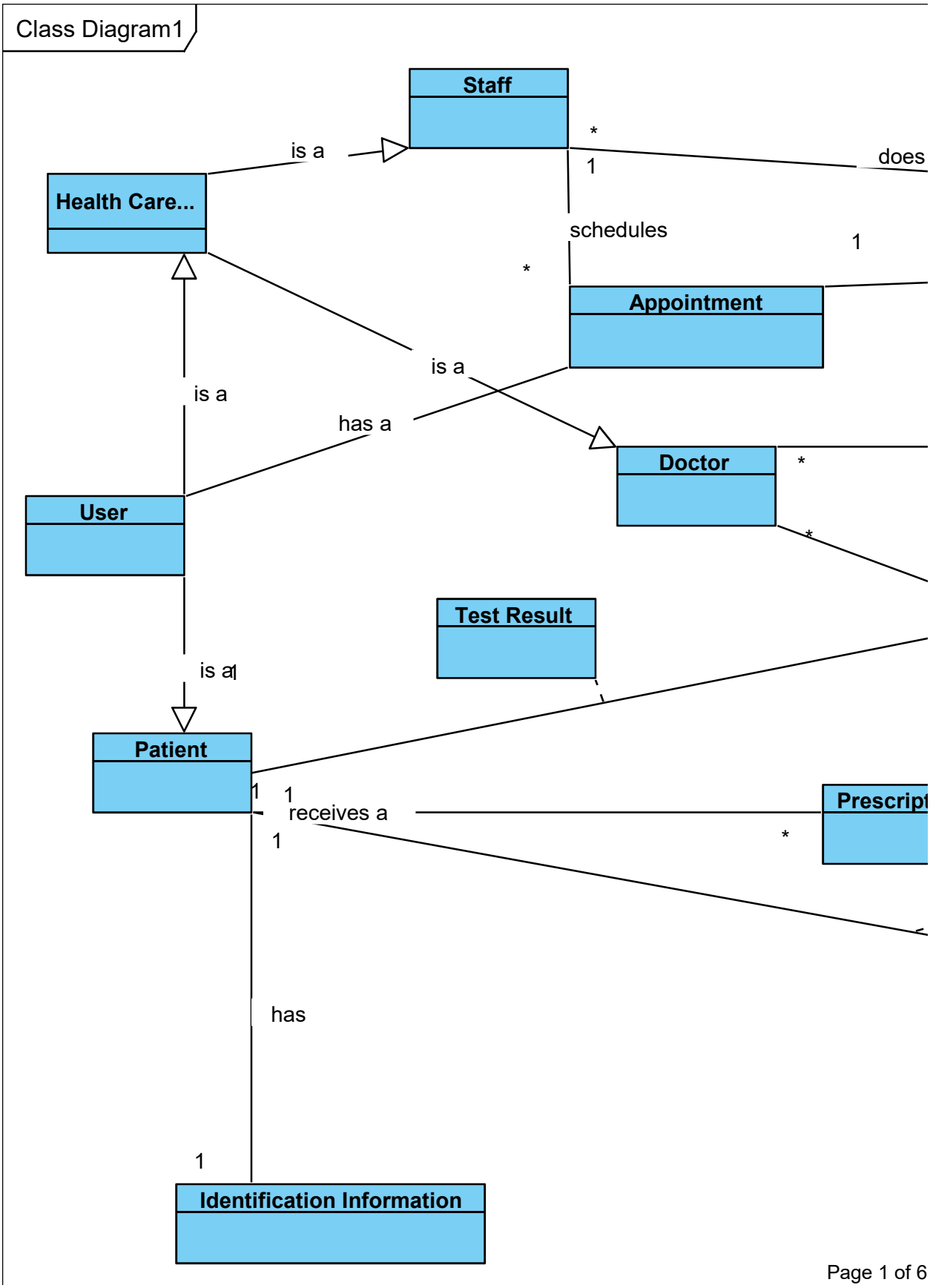
Requirements

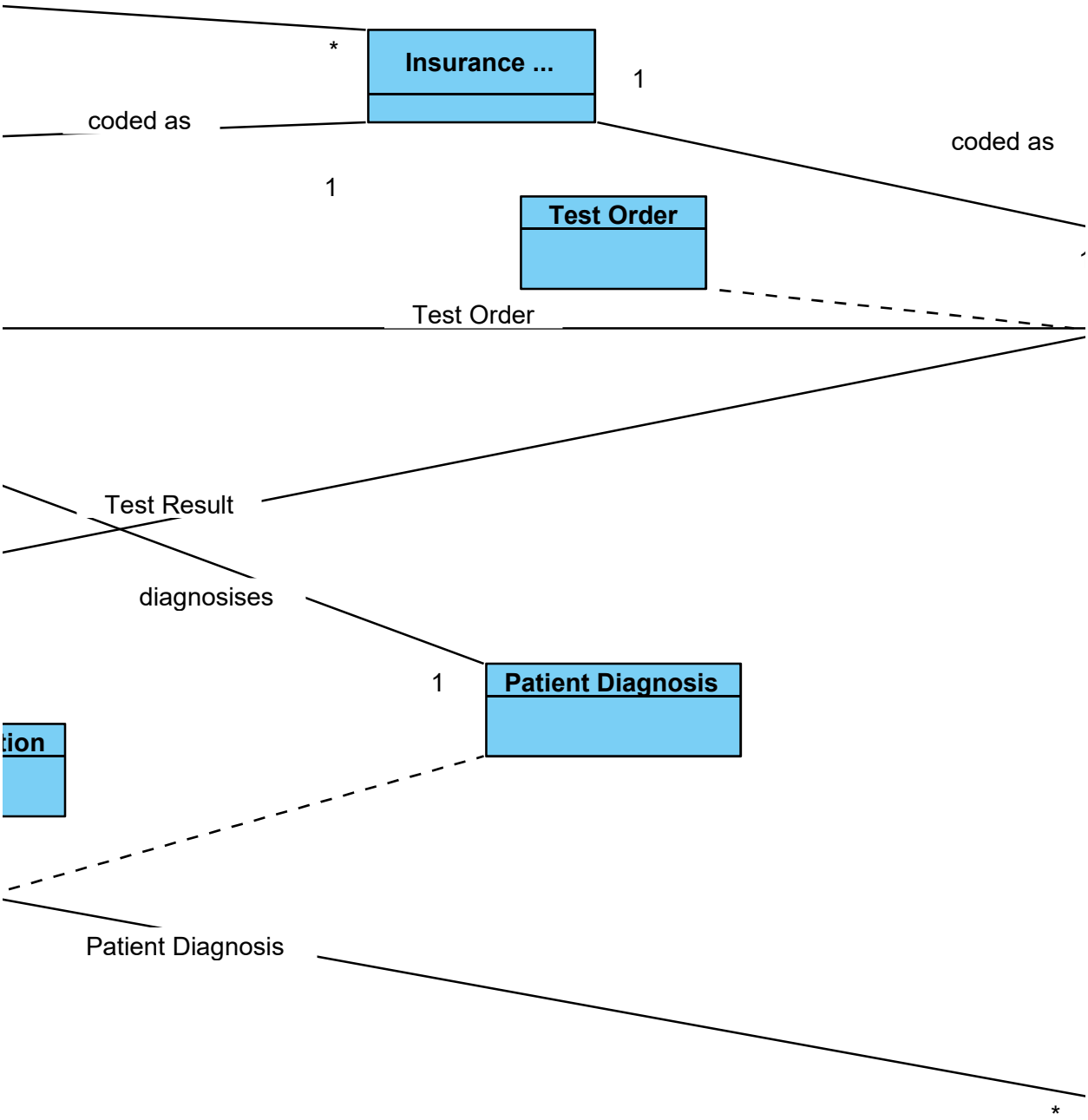
Window with fields for patient data followed by text boxes to enter information

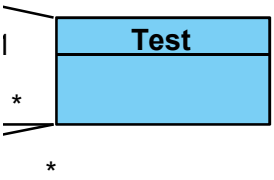
Different fields for different types of patient visits

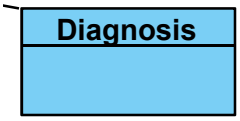
Doctor-to-Patient

Time sheet


Class Diagram1
Staff
is a
*
1
does
Health Care...
schedules
1
*
Appointment
is a
is a
has a
Doctor
*
User
*
Test Result
is a
Patient
1
1
receives a
Prescript
1
*
has
1
Identification Information




Test
*
*


Diagnosis


Community Edition

req SE1 project

<<requirement>>
Different fields for different types of patient visits

Text = "All patients must be accounted for
- correct info for each patient (database)"
ID = "REQ001"
source = ""
kind = ""
verifyMethod = ""
risk = ""
status = ""

<<requirement>>

Text = "Doctors have to be able to use th
interface"
ID = "REQ002"
source = ""
kind = ""
verifyMethod = ""
risk = ""
status = ""

<<requirement>>
Doctor Database

Text = "Doctor Information
- patients seeing"
ID = "REQ004"
source = ""
kind = ""
verifyMethod = ""
risk = ""
status = ""

<<requirement>>
Time sheet

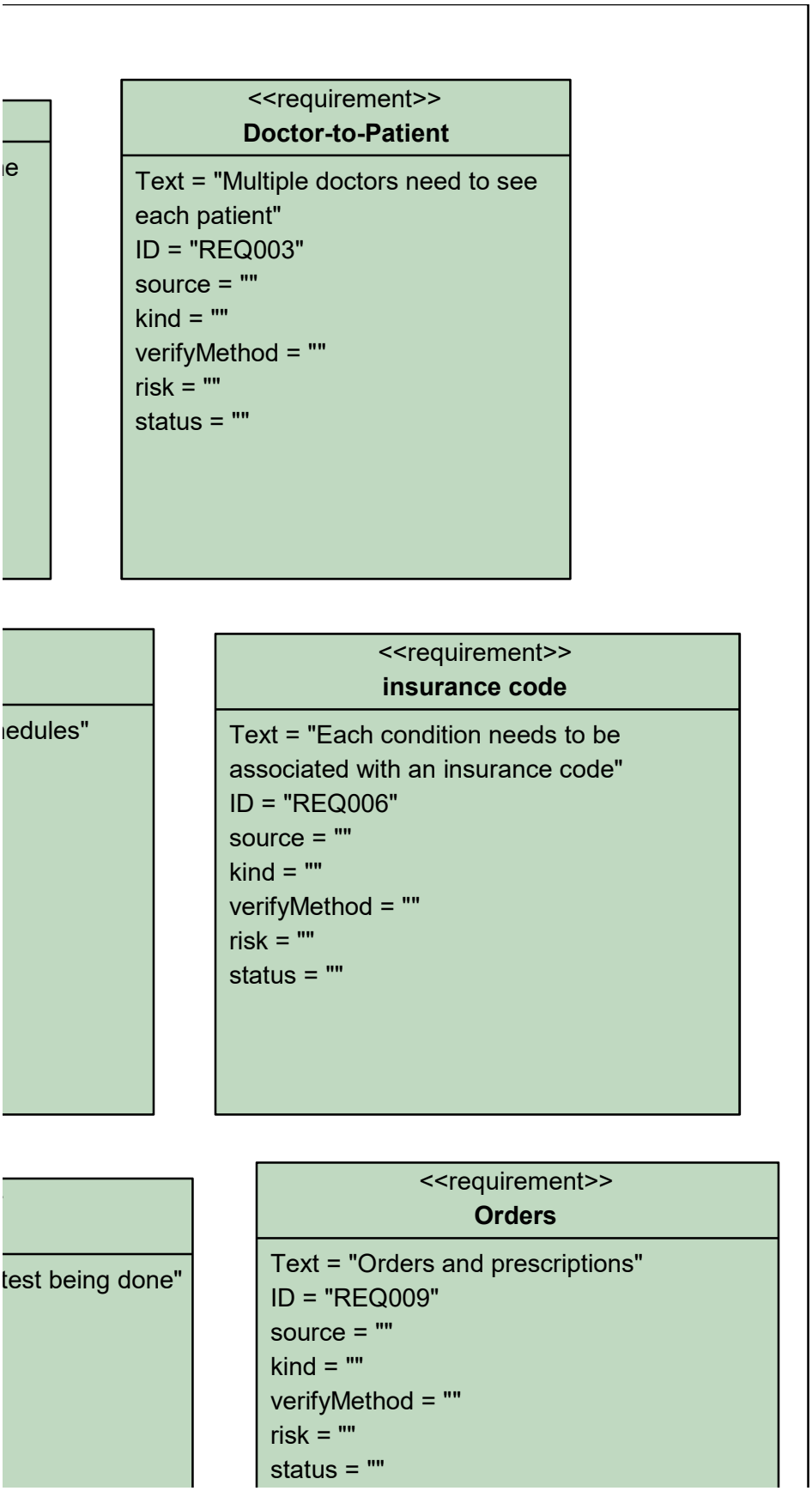
Text = "Time sheet for doctors sch
ID = "REQ005"
source = ""
kind = ""
verifyMethod = ""
risk = ""
status = ""

<<requirement>>
Patient - User Interface

Text = ""
ID = "REQ007"
source = ""
kind = ""
verifyMethod = ""
risk = ""
status = ""

<<requirement>>
Clinical Test

Text = "Orders and view clinical
ID = "REQ008"
source = ""
kind = ""
verifyMethod = ""
risk = ""
status = ""


<<requirement>>
Doctor-to-Patient
Text = "Multiple doctors need to see each patient"
ID = "REQ003"
source = ""
kind = ""
verifyMethod = ""
risk = ""
status = ""
<<requirement>>
insurance code
Text = "Each condition needs to be associated with an insurance code"
ID = "REQ006"
source = ""
kind = ""
verifyMethod = ""
risk = ""
status = ""
<<requirement>>
Orders
Text = "Orders and prescriptions"
ID = "REQ009"
source = ""
kind = ""
verifyMethod = ""
risk = ""
status = ""

risk
status = ""

risk
status = ""

<<requirement>>
Look up

Text = "Fast lookup"
ID = "REQ010"
source = ""
kind = ""
verifyMethod = ""
risk = ""
status = ""

<<requirement>>
Access

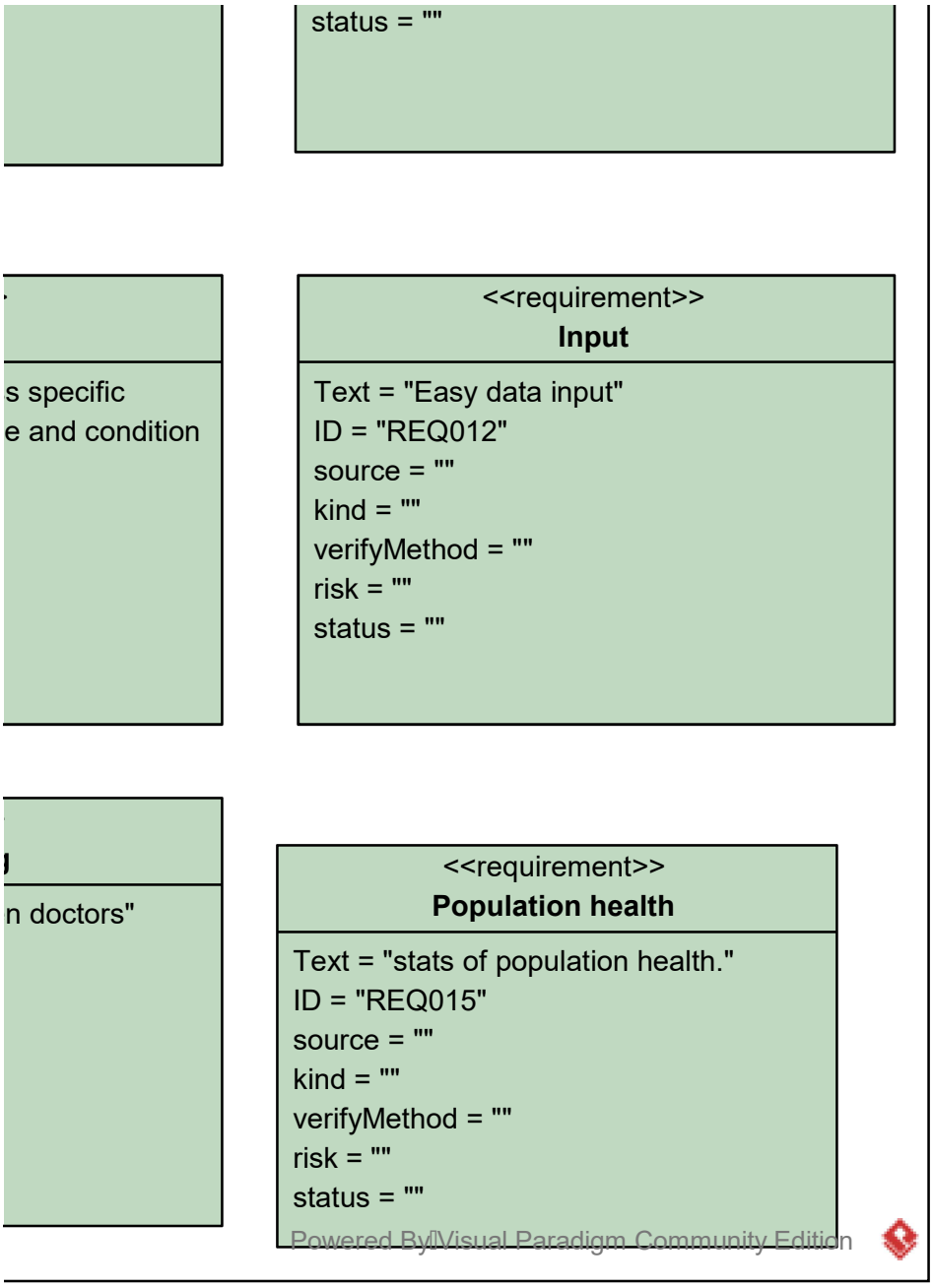
Text = "System should be acces
(should only display patient nam
to doctors)"
ID = "REQ011"
source = ""
kind = ""
verifyMethod = ""
risk = ""
status = ""

<<requirement>>
Data consistency

Text = "Managing data"
ID = "REQ013"
source = ""
kind = ""
verifyMethod = ""
risk = ""
status = ""

<<requirement>>
Easy file sharing

Text = "Easy file sharing betwee
ID = "REQ014"
source = ""
kind = ""
verifyMethod = ""
risk = ""
status = ""


status = ""
s specific
e and condition
<<requirement>>
Input
Text = "Easy data input"
ID = "REQ012"
source = ""
kind = ""
verifyMethod = ""
risk = ""
status = ""
n doctors"
<<requirement>>
Population health
Text = "stats of population health."
ID = "REQ015"
source = ""
kind = ""
verifyMethod = ""
risk = ""
status = ""
Powered By Visual Paradigm Community Edition