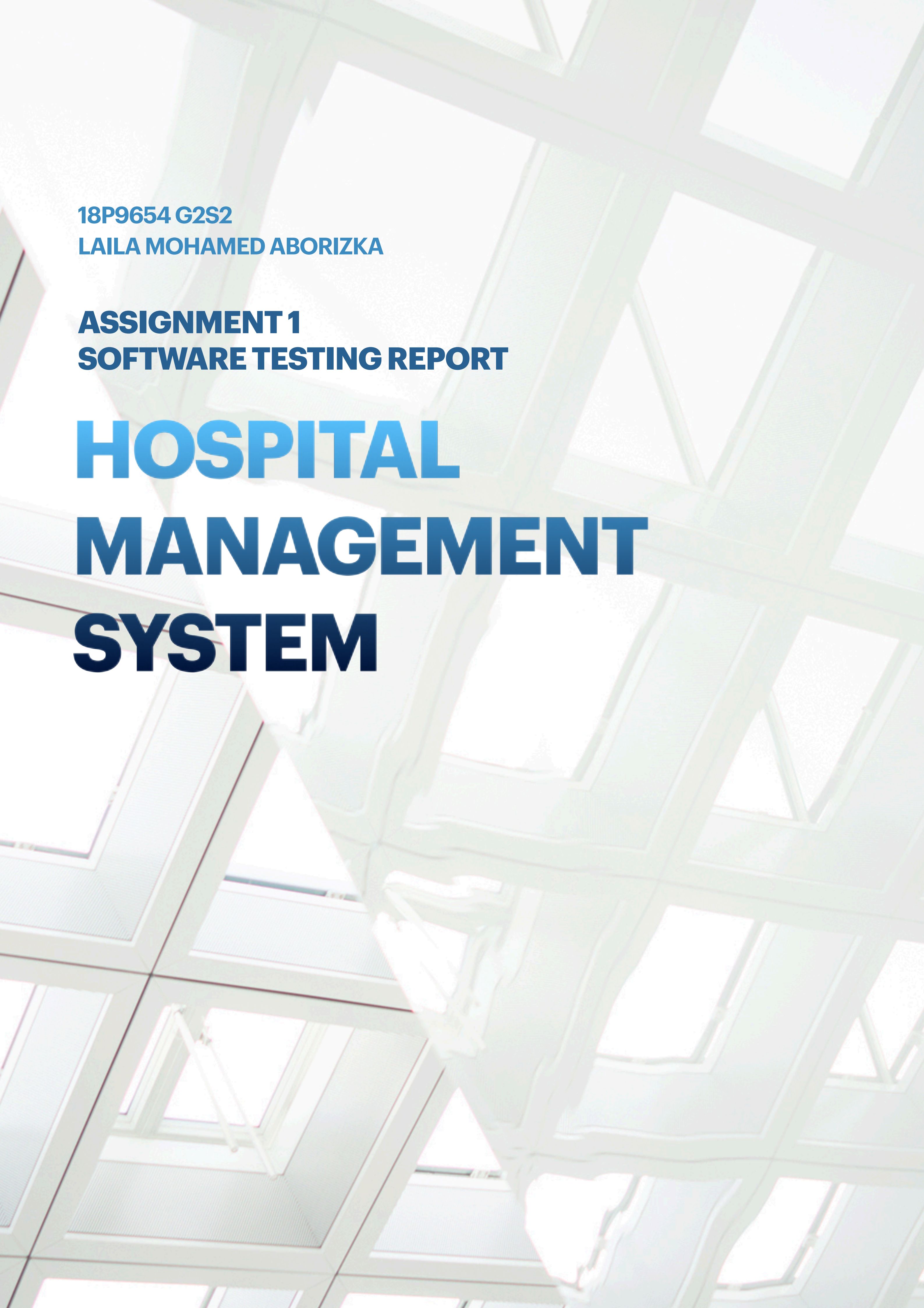


18P9654 G2S2
LAILA MOHAMED ABORIZKA

ASSIGNMENT 1
SOFTWARE TESTING REPORT

HOSPITAL MANAGEMENT SYSTEM

QUESTION ONE #246394

Functional testing starts with breaking down the specification into independently testable features.

TESTABLE FEATURES

Registration

Add patients The front-desk staff adds new patients to the system

Assign ID The front-desk staff gives each patient a ID and add it to the patient's record.

Consultation

Assign Ward The consulting nurse assigns patients to an appropriate ward.

Assign to WaitingList If no bed is available the consulting nurse adds patients on the waiting list.

Medical matter management

Assign Doctor The administrative staff in the ward assigns a doctor to a given patient.

Surgery case In a surgery case, the administrative staff assigns a surgery room, surgeon and nurses to the patient.

Check Out

Delete Patient ID The administrative staff in the ward deletes the ID of the patient from the system when the patient checks out.

Report Generation

Bed Availability Every six hours the system shall generate reports on bed availability.

Staff Schedule Every six hours the system shall generate reports on staff schedule.

Database

Search forPatient User searches for patient's information by last name or PHN or patient ID.

Update Patient Information The system shall allow the user to update any oft he patient's information.

QUESTION TWO, THREE, FOUR

Since there is no knowledge of any internal implementation details (no code provided) all of the testing techniques applied are black-box testing techniques.

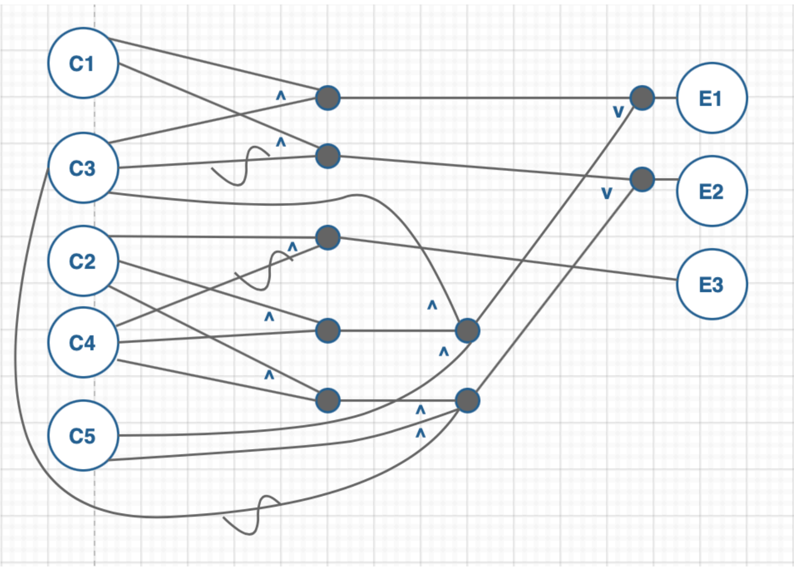
Registration

Add patients

Testing technique Patient may be registered in hospital and may be not . (Assume PHN is a 4 figure number and that all the personal details are saved with the PHN)

To test this function cause-effect technique shall be used:

Causes: C1: patient is registered
C2: patient is not registered
C3: PHN is entered
C4: PHN is assigned
C5: Data is entered
Effects: E1: Patient is added
E2: Print “PHN not entered”
E3: Print “PHN needs to be assigned”



C1	1	X	1	X	X
C2	X	1	X	1	1
C3	1	1	0	0	X
C4	X	1	X	1	0
C5	X	1	X	1	X
E1	1	1			
E2			1	1	
E3					1

Testing requirements

Preconditions

Validated

PHN is entered when patient is registered.
PHN is assigned then entered when patient is not registered.

Patient either registered or not.

Assumed

Staff is front-desk.

A list of registered patients in hospital with their personal info exists.

If PHN assigned data is entered and is valid (Address is string, date in format DD-MM-YY, Telephone No. Is 10 digits and starts with +02) (GUI ensures this).

Postconditions

If patient is registered —> PHN is entered into the computer and a message displays successful admission.

If patient is not registered —> A new PHN is assigned and entered into the computer together with Date of Birth, Address,Telephone No. and a message displays successful admission.

If patient is registered and PHN is not entered —> an error message is displayed.

If patient is not registered and PHN is not assigned —> an error message is displayed.

If patient is not registered and PHN is assigned but not entered—> an error message is displayed.

Requirements

system needs to be tested with patient registered and PHN entered.

system needs to be tested with patient registered and PHN not entered.

system needs to be tested with patient not registered and PHN assigned, entered and data entered.

system needs to be tested with patient not registered and PHN not assigned.

system needs to be tested with patient not registered and PHN assigned, but not entered and data entered.

Testing specifications

Test 1: registered, PHN entered: 1234

upon return expect: successful admission

Test 2: registered, PHN entered: ____

upon return expect: error: "PHN missing"

Test 3: Not registered, PHN assigned: 6789, 6789(entered), Address: "70 A Cairo", Date:12-06-20, Telephone:+0237485673

upon return expect: successful admission

Test 4: Not registered, PHN assigned: ___

upon return expect: error: "PHN not assigned"

Test 5: Not registered, PHN assigned: 6789, ____(NOT entered), Address: "70 A Cairo", Date:12-06-20, Telephone:+0237485673

upon return expect: error: "PHN missing"

Assign ID

The front-desk staff gives each patient a ID and add it to the patient's record. The ID maybe valid or invalid in terms of format (8 figure number ends with a 0).

The ID may exist in system or not.

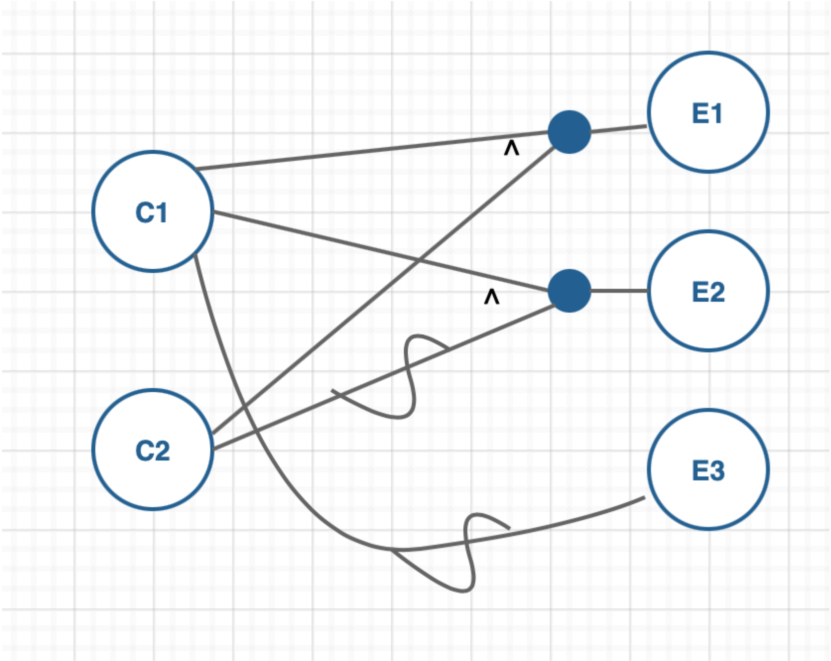
Causes : C1: ID valid

C2: ID does not exist in system

Effects: E1: ID assigned

E2: Error message : "ID already exists"

E3: Error message : "ID invalid"



C1	1	1	0
C2	1	0	X
E1	1		
E2		1	
E3			1

Testing requirements

Preconditions

Validated ID Valid

ID doesn't exist in database of patients (unique)

Assumed Staff is front-desk.

A list of already existing IDs of patients exists.

Postconditions

If ID is valid and doesn’t exist in system —> ID assigned and added to patient’s record.

If ID is valid and exists in system —> an error message is displayed ”ID already exists”.

If ID is not valid —> an error
message is displayed ”ID already exists”.

Requirements

system needs to be tested with ID valid and not in system.

system needs to be tested with ID valid and already exists in system.

system needs to be tested with ID not valid

Testing specifications

Test 1: ID inserted: 12345670

upon return expect: ID successfully assigned

Test 2: ID inserted: 76543210

upon return expect: error: “ID already exists”

Test 3: ID inserted: 123A0

upon return expect: error: “ID not valid”

Consultation

Assign Ward	Ward: Maternity Surgical Cancer Cardiac Condition: Maternity_Cond Surgical_Cond Cancer_Cond Cardiac_Cond Bed: Available Not available

Pairwise Testing:

Ward	Condition	Bed
Maternity	Maternity_Cond	Available
Surgical	Surgical_Cond	Not Available
Cancer	Cancer_Cond	Available
Cardiac	Cardiac_Cond	Available
Maternity	Cancer_Cond	Not Available
Surgical	Maternity_Cond	Available
Cancer	Surgical_Cond	Not Available
Cardiac	Surgical_Cond	~Not Available
Maternity	Surgical_Cond	Available
Surgical	Cardiac_Cond	Not Available
Cancer	Cardiac_Cond	~Available
Cardiac	Maternity_Cond	Not Available
Maternity	Cardiac_Cond	~Available
Surgical	Cancer_Cond	Not Available
Cancer	Maternity_Cond	~Available
Cardiac	Cancer_Cond	~Not Available

Testing requirements

Preconditions

Validated

Patient condition

Nurse decision of which ward the patient needs

Bed status

Assumed

Staff is nurse

A ward that satisfies the condition exists in the hospital.

Postconditions

If condition is assigned the correct ward and bed is available —> patient assigned successfully

If condition is assigned the correct ward and bed is not available —> message : “add patient to waiting list”

If condition is not assigned the correct ward and bed is not available —> incorrect assignment of patient

If condition is not assigned the correct ward and bed is available —> incorrect assignment of patient

Requirements

system needs to be tested with each ward assigned to each condition.

system needs to be tested with bed available with each ward

system needs to be tested with bed available with each condition

system needs to be tested with bed not available with each ward

system needs to be tested with bed not available with each condition

Testing specifications

Test 1: Ward: Maternity, Condition: Maternity, Bed: Available
upon return expect: patient assigned successfully

Test 2: Ward: Surgical, Condition: Surgical, Bed:Not Available
upon return expect: message : “add patient to waiting list”

Test 3: Ward: Cancer, Condition: Cancer, Bed: Available
upon return expect: patient assigned successfully

Test 4: Ward: Cardiac, Condition: Cardiac, Bed: Available
upon return expect: patient assigned successfully

Test 5: Ward: Maternity, Condition: Cancer, Bed:Not Available
upon return expect: incorrect assignment of patient

Test 6: Ward: Surgical, Condition: Maternity, Bed: Available
upon return expect: incorrect assignment of patient

Test 7: Ward: Cancer, Condition: Surgical, Bed:Not Available
upon return expect: incorrect assignment of patient

Test 8: Ward: Cardiac, Condition: Surgical, Bed: Available
upon return expect: incorrect assignment of patient

Test 9: Ward: Maternity, Condition: Surgical, Bed: Available
upon return expect: incorrect assignment of patient

Test 10: Ward: Surgical, Condition: Cardiac, Bed:Not Available
upon return expect: incorrect assignment of patient

Test 11: Ward: Cancer, Condition: Cardiac, Bed: Available
upon return expect: incorrect assignment of patient

Test 12: Ward: Cardiac,Condition: Maternity, Bed:Not Available
upon return expect: incorrect assignment of patient

Test 13: Ward: Maternity, Condition: Cardiac, Bed: Available
upon return expect: incorrect assignment of patient

Test 14: Ward: Surgical, Condition: Cancer, Bed:Not Available
upon return expect: incorrect assignment of patient

Test 15: Ward: Cancer, Condition: Maternity, Bed: Available
upon return expect: incorrect assignment of patient

Test 16: Ward: Cardiac, Condition: Cancer, Bed: Available
upon return expect: incorrect assignment of patient

Assign to WaitingList If no bed is available the consulting nurse adds patients on the waiting list. Here it is assumed that emergency cases are assigned beds immediately (the hospital has a reserve for emergencies).

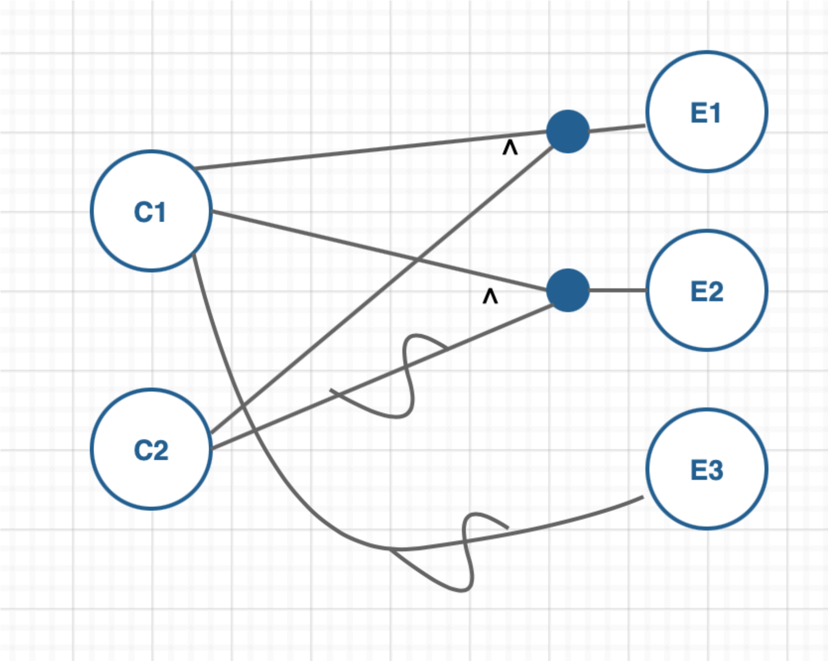
Causes: C1: Bed is not available

C2: Not an emergency

Effects: E1: Add to waiting list

E2: Error message “Case is an emergency”

E3: Error message “A bed is available”



C1	1	1	0
C2	1	0	X
E1	1		
E2		1	
E3			1

Testing requirements

Preconditions

Validated Bed Status

Emergency Status

Assumed Staff is nurse.

Waiting list has no limit.

Postconditions

If bed is available —> Error message “A bed is available”

If bed is not available and case is emergency —>Error message “Case is an emergency”

If bed is not available and case is not emergency. —>waiting list updated successfully.

Requirements

system needs to be tested with bed available

system needs to be tested with bed not available but case is emergency.

system needs to be tested with bed not available and case is not an emergency.

Testing specifications

Test 1: Bed: Available, Case: Not an emergency

> upon return expect:Error message “A bed is available”

Test 2: Bed: Not Available, Case: emergency

> upon return expect: Error message “Case is an emergency”

Test 3: Bed: Not Available, Case: Not emergency

> upon return expect: waiting list updated successfully.

Medical matter management

Surgery case In a surgery case, the administrative staff assigns a surgery room, surgeon and nurses to the patient.

Case:

Surgery

Not surgery

Surgery Room :

Available

Not available

Surgeon:

Available

Not available

Nurse:

Available

Not available

Pairwise Testing:

Case	Surgery Room	Surgeon	Nurse
Surgery	Available	Available	Available
Not surgery	Available	Available	Available
Surgery	Not available	Not available	Not available
Not surgery	Not available	Not available	Not available
Surgery	Available	Not available	Available
Not surgery	Not available	Available	Available
Surgery	Available	Not available	Not available
Not surgery	Not available	Available	Not available

Testing requirements

Preconditions

Validated

Case

Surgery Room Status

Surgeon Status

Nurse Status

Assumed

Staff is administrative staff

A list with cases that need a surgery exists in database.

Postconditions

If case is surgery and room, surgeon, and nurse are available —> surgery is ready

If case is not a surgery —> Error: case doesn't need a surgery

If case is surgery and room is not available —> message: wait for a room to be available

If case is surgery and surgeon is not available —> message: wait for a surgeon to be available

If case is surgery and nurse is not available —> message: wait for a nurse to be available

Requirements

system needs to be tested with case not surgery

system needs to be tested with case surgery and all available

system needs to be tested with case surgery room not available

system needs to be tested with case surgery and surgeon not available

system needs to be tested with case surgery and nurse not available

Testing specifications

Test 1: Case: Surgery, Surgery_Room: Available, Surgeon: Available, Nurse: Available

upon return expect: surgery is ready

Test 2: Case:Not a Surgery, Surgery_Room: Available, Surgeon: Available, Nurse: Available

upon return expect: Error: case doesn't need a surgery

Test 3: Case: Surgery, Surgery_Room:Not Available, Surgeon: Not Available, Nurse:Not Available

upon return expect: message: wait for a room to be available

message: wait for a surgeon to be available

message: wait for a nurse to be available

Test 4: Case:Not a Surgery, Surgery_Room: Not Available, Surgeon: Not Available, Nurse: Not Available

upon return expect: Error: case doesn't need a surgery

Test 5: Case: Surgery, Surgery_Room: Available, Surgeon: Not Available, Nurse: Available

upon return expect: message: wait for a surgeon to be available

Test 6: Case:Not Surgery, Surgery_Room:Not Available, Surgeon: Available, Nurse: Available

upon return expect: message: wait for a surgeon to be available

message: wait for a nurse to be available

Test 7: Case: Surgery, Surgery_Room: Available, Surgeon: Not Available, Nurse:Not Available

upon return expect: incorrect assignment of patient

Test 8: Case:Not Surgery, Surgery_Room:Not Available, Surgeon: Available, Nurse:Not Available

upon return expect: Error: case doesn't need a surgery

Check Out

Delete Patient ID

The administrative staff in the ward deletes the ID of the patient from the system when the patient checks out. It is assumed that in order to delete the ID it has to exist in system and there has to be a deletion request to ensure accurate handling of checkout.

Causes: C1: Patient checks out

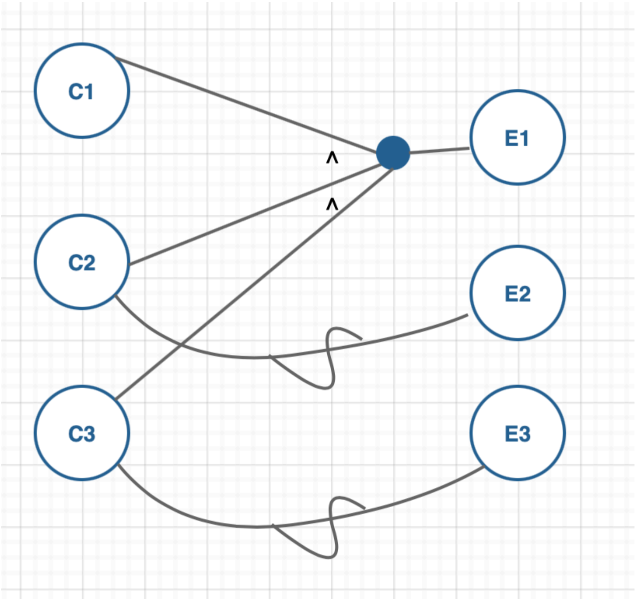
C2: Delete patient ID request

C3: ID exists in system

Effects: E1: The patient ID is deleted

E2: Error message “Request required”

E3: Error message “ID doesn’t exist”



C1	1	X	X
C2	1	0	X
C3	1	X	0
E1	1		
E2		1	
E3			1

Testing requirements

Preconditions

Validated Patient checkout
Request

ID exists in system

Assumed Staff is administrative staff.

A list of all patients IDs is available .

Postconditions

If request is not available —> Error message "Request required"

If ID does not exist in system—>Error message "ID doesn't exist"

If patient checks out and request is available and ID is found in system —> The patient ID is deleted

Requirements

system needs to be tested with request not available

system needs to be tested with ID not existing

system needs to be tested with patient checkout and request available and ID exists

Testing specifications

Test 1: Patient: checkout, Request:available, ID: 12345670(exists)

upon return expect: The patient ID is deleted

Test 2: Patient: checkout, Request:available, ID: 76554670(doesn't exist)

upon return expect: Error message "ID doesn't exist"

Test 3: Patient: checkout, Request:Not available, ID: 76554670(doesn't exist)

upon return expect: Error message "Request required"

Error message "ID doesn't exist"

Report Generation

Bed Availability

Ward:

Maternity
Surgical
Cancer
Cardiac

Bed number:(assume each ward has 4 beds)
One
Two
Three
Four

Bed:
Occupied
Not occupied

Pairwise Testing:

Ward	Bed No.	Bed
Maternity	One	Occupied
Surgical	Two	Not Occupied
Cancer	Three	Occupied
Cardiac	Four	Occupied
Maternity	Three	Not Occupied
Surgical	One	Occupied
Cancer	Two	Not Occupied
Cardiac	Two	~Not Occupied
Maternity	Two	Occupied
Surgical	Four	Not Occupied
Cancer	Four	~Occupied
Cardiac	One	Not Occupied
Maternity	Four	~Occupied
Surgical	Three	Not Occupied
Cancer	One	~Occupied
Cardiac	Three	~Not Occupied

Testing requirements

Preconditions

Validated

Ward Type

Bed number

Bed status

Assumed

Report generation is working correctly and generating reports every 6 hours

Postconditions

If Bed is occupied —> report prints : Ward Type, Bed No., Bed: Occupied

If Bed is not occupied —> report prints : Ward Type, Bed No., Bed: Not Occupied

Requirements

system needs to be tested with each ward Type

system needs to be tested with each bed in each ward

system needs to be tested with bed occupied

system needs to be tested with bed not occupied

Testing specifications

Test 1: Ward: Maternity, Bed_no: One, Bed: Occupied

upon return expect: report prints : Ward Type: Maternity, Bed No.:One, Bed: Occupied

Test 2: Ward: Surgical, Bed_no: Two, Bed:Not Occupied

upon return expect: report prints : Ward Type: Surgical, Bed No.:Two, Bed:Not Occupied

Test 3: Ward: Cancer, Bed_no: Three, Bed: Occupied

upon return expect: report prints : Ward Type: Cancer, Bed No.: Three, Bed: Occupied

Test 4: Ward: Cardiac, Bed_no: Four, Bed: Occupied

upon return expect: report prints : Ward Type: Cardiac, Bed No.: Four, Bed: Occupied

Test 5: Ward: Maternity, Bed_no: Three, Bed:Not Occupied

upon return expect:report prints : Ward Type: Maternity, Bed No.:Three, Bed:Not Occupied

Test 6: Ward: Surgical, Bed_no: One, Bed: Occupied

upon return expect: report prints : Ward Type: Surgical, Bed No.:One, Bed: Occupied

Test 7: Ward: Cancer, Bed_no: Two, Bed:Not Occupied

upon return expect: report prints : Ward Type: Cancer, Bed No.: Two, Bed:Not Occupied

Test 8: Ward: Cardiac, Bed_no: Two, Bed: Not Occupied

upon return expect: report prints : Ward Type: Cardiac, Bed No.: Two, Bed: Not Occupied

Test 9: Ward: Maternity, Bed_no: Two, Bed: Occupied

upon return expect: report prints : Ward Type: Maternity, Bed No.: Two, Bed: Occupied

Test 10: Ward: Surgical, Bed_no: Four, Bed:Not Occupied

upon return expect: report prints : Ward Type: Surgical, Bed No.: Four, Bed:Not Occupied

Test 11: Ward: Cancer, Bed_no: Four, Bed: Occupied

upon return expect: report prints : Ward Type: Cancer, Bed No.: Four, Bed: Occupied

Test 12: Ward: Cardiac, Bed_no: One, Bed:Not Occupied

upon return expect: report prints : Ward Type: Cardiac, Bed No.: One, Bed:Not Occupied

Test 13: Ward: Maternity, Bed_no: Four, Bed: Occupied

upon return expect: report prints : Ward Type: Maternity, Bed No.: Four, Bed: Occupied

Test 14: Ward: Surgical, Bed_no: Three, Bed:Not Occupied

upon return expect: report prints : Ward Type: Surgical, Bed No.: Three, Bed:Not Occupied

Test 15: Ward: Cancer, Bed_no: One, Bed: Occupied

upon return expect: report prints : Ward Type: Cancer, Bed No.:One, Bed: Occupied

Test 16: Ward: Cardiac, Bed_no: Three, Bed: Not Occupied

upon return expect: report prints : Ward Type: Cardiac, Bed No.: Three, Bed:Not Occupied

Staff Schedule

Staff_ID:

Valid
Invalid

Staff Name :
Exists
Doesn’t exist

Staff Type:
Doctor
Nurse
Administrator
Front-desk staff

Dutyshift:
Day
Night

Pairwise Testing:

Staff_ID	Staff_Name	Staff_type	Dutyshift
Valid	Exists	Doctor	Day
Invalid	Exists	Nurse	Day
Valid	Doesn’t exist	Administrator	Night
Invalid	Doesn’t exist	Front-desk	Night
Valid	~Exists	Nurse	~Day
Invalid	Doesn’t exist	Doctor	Day
Valid	Exists	Front-desk	Night
Invalid	~Doesn’t exist	Administrator	~Night
~Valid	Exists	Administrator	Day
~Invalid	Doesn’t exist	Nurse	Night
~Valid	~Exists	Front-desk	Day
~Invalid	~Doesn’t exist	Doctor	Night

Testing requirements

Preconditions

Validated

Valid Staff_ID

Staff member name exists

Staff_type

Dutyshift

Assumed

Report generation is working correctly and generating reports every 6 hours

The name is valid (valid doesn't necessarily mean existing)

Staff insert their IDs to start and finish shifts

Postconditions

If ID is invalid —> no update in report generated

If ID is valid and staff_Name exists —> report prints : name

If ID is valid and staff_Name doesn't exist —> no update in report generated

If ID is valid —> report prints : Staff type associated with ID

If ID is valid and it is Dayshift —> report prints : Day

If ID's valid and it's Nightshift—>report prints : Night

Requirements

system needs to be tested with invalid ID

system needs to be tested with valid ID

system needs to be tested with valid ID and staff_Name exists

system needs to be tested with valid ID and staff_Name does not exist

system needs to be tested with Dayshift

system needs to be tested with Nightshift

system needs to be tested with all staff_types

Testing specifications

Test 1: Staff_ID: 1234 (Valid), Staff_Name: Laila (exists), Staff_type: Doctor, Duty-shift: Day

upon return expect: report prints : Staff_ID: 1234, Staff_Name: Laila , Staff_type: Doctor, Duty-shift: Day

Test 2: Staff_ID: 12B4 (InValid), Staff_Name: Laila (exists), Staff_type: Nurse, Duty-shift: Day

upon return expect: no update in report generated

Test 3: Staff_ID: 1234 (Valid), Staff_Name: Anna (doesn't exist), Staff_type: Administrator, Duty-shift: Night

upon return expect: report prints : no update in report generated

Test 4: Staff_ID: 12B4 (InValid), Staff_Name: Anna (doesn't exist), Staff_type: Front-desk, Duty-shift: Night

upon return expect: no update in report generated

Test 5: Staff_ID: 1234 (Valid), Staff_Name: Laila (exists), Staff_type: Nurse, Duty-shift: Day

upon return expect: report prints : Staff_ID: 1234, Staff_Name: Laila , Staff_type: Nurse, Duty-shift: Day

Test 6: Staff_ID: 12B4 (InValid), Staff_Name: Anna (doesn't exist), Staff_type: Doctor, Duty-shift: Day

upon return expect: no update in report generated

Test 7: Staff_ID: 1234 (Valid), Staff_Name: Laila (exists), Staff_type: Front-desk, Duty-shift: Night

upon return expect: report prints : Staff_ID: 1234, Staff_Name: Laila , Staff_type: Front-desk, Duty-shift: Night

Test 8: Staff_ID: 12B4 (InValid), Staff_Name: Anna (doesn't exist), Staff_type: Administrator, Duty-shift: Night

upon return expect: no update in report generated

Test 9: Staff_ID: 1234 (Valid), Staff_Name: Laila (exists), Staff_type: Administrator, Duty-shift: Day upon return expect: report prints : Staff_ID: 1234, Staff_Name: Laila , Staff_type: Administrator, Duty-shift: Day

Test 10: Staff_ID: 12B4 (InValid), Staff_Name: Anna (doesn't exist), Staff_type: Nurse, Duty-shift: Night

upon return expect: no update in report generated

Test 11: Staff_ID: 1234 (Valid), Staff_Name: Laila (exists), Staff_type: Front-desk, Duty-shift: Day

upon return expect: report prints : Staff_ID: 1234, Staff_Name: Laila , Staff_type: Front-desk, Duty-shift: Day

Test 12: Staff_ID: 12B4 (InValid), Staff_Name: Anna (doesn't exist), Staff_type: Doctor, Duty-shift: Night

upon return expect: no update in report generated

Database

Update Patient Information

Causes:

C1: Patient found

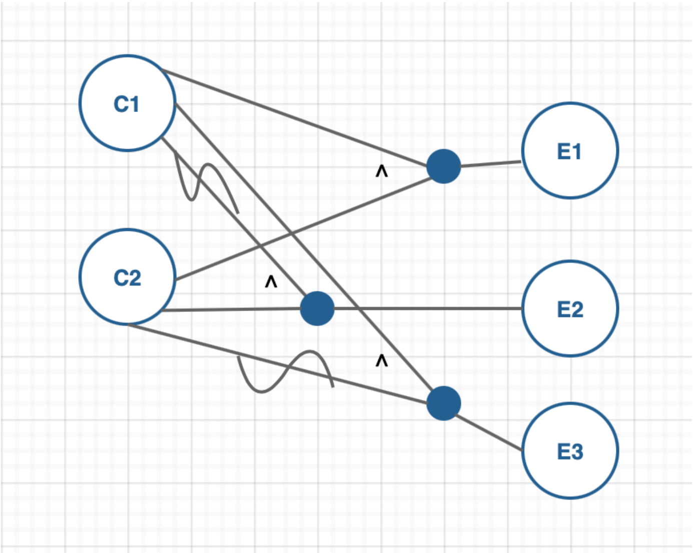
C2: New data Valid

Effects:

E1: Info updated

E2:Error message "Patient not found"

E3:Error message "Data Invalid"



C1	1	0	X
C2	1	X	0
E1	1		
E2		1	
E3			1

Testing requirements

Preconditions

Validated	Patient found
	New data valid
Assumed	Database contains all patients’ info

Postconditions

If patient is found and new data is valid —> Info updated

If patient is found and new data is invalid —> Error message “Data Invalid”

If patient is not found —>Error message “Patient not found”

Requirements

system needs to be tested with patient found and data valid

system needs to be tested with patient found and data invalid

system needs to be tested with patient not found

Testing specifications

Test 1: Patient: 1234 (PHN found), Data:valid

upon return expect: Info updated

Test 2: Patient: 0224 (PHN not found), Data:valid

upon return expect: Error message “Patient not found”

Test 3: Patient: 1234 (PHN found), Data:invalid

upon return expect: Error message “Data Invalid”

Search forPatient

User searches for patient's information by last name or PHN or patient ID.

Assume last name has to be a string

and patient ID is made of 8 figures and ends with a

zero (xxxxxxx0) and PHN is made of 4 figures (xxxx)

Equivalence partitioning :

Input	Valid classes	Invalid classes
Last name	1 word that is a string all letters	Numbers String with symbols String with whitespace Empty Value
ID	8 figure number ends with a 0 e.g 12345670	More than 8 figures Less than 8 figures Malformed numbers Empty Value Invalid Format String
PHN	4 figure number	More than 4 figures Less than 4 figures Malformed numbers Empty Value String

Testing requirements

Preconditions

Validated Last Name valid

ID valid

PHN valid

Assumed Database contains all patients' info

Postconditions

If last name is valid and exists —> patient found

If last name is valid and does not exist —> patient not found

If last name is invalid —> Error message

If ID is valid and exists —> patient found

If ID is valid and does not exist —> patient not found

If ID is invalid —> Error message

If PHN is valid and exists —> patient found

If PHN is valid and does not exist —> patient not found

If PHN is invalid —> Error message

Requirements

system needs to be tested with ID invalid

system needs to be tested with PHN invalid

system needs to be tested with Last_Name invalid

system needs to be tested with ID valid and exists

system needs to be tested with PHN valid and exists

system needs to be tested with Last_Name valid and exists

system needs to be tested with ID valid and doesn't exist

system needs to be tested with PHN valid and doesn't exist

system needs to be tested with Last_Name valid and doesn't exist

Testing specifications

Test 1: ID: 123

upon return expect: Error message

Test 2: Last_Name: 123

upon return expect: Error message

Test 3: PHN: 123

upon return expect: Error message

Test 4: ID: 12345670 (this exists)

upon return expect: patient found

Test 5: Last_Name: Mohamed (this exists)

upon return expect: patient found

Test 6: PHN: 1234 (this exists)

upon return expect: patient found

Test 7: ID: 76543210 (this doesn’t exist)

upon return expect: patient not found

Test 8: Last_Name: Ahmed (this doesn’t exist)

upon return expect: patient not found

Test 9: PHN: 1237 (this doesn’t exist)

upon return expect: patient not found