ASSIGNMENT#3

Case Study

NOVEMBER 2, 2020

BSE181053_HAMZA SHAHZAD
BSE181056_FAHAD ALI

Case Study: Online Clinic workflow:

Introduction:

This system will be designed to improve clinical workflow. It will connect patient and clinic online. Nowadays no one has time to visit clinic and wait for appointment. This system will help for getting online appointments. Patient can get appointment through Internet.

The purpose of this document is to provide patients this facility that they can get appointment from their home. They do not have to visit clinic just to get appointment. Everyone is busy now a days, no one has time to visit clinic for appointment. As well as patient's medical history will be saved on this system so every time when patient visit doctor so he/she does not have to bring medical report along or doctor does not have to check B.P or weight etc. The software includes maintaining patient details, provide prescriptions, precautions and diet advice. Providing and maintaining all kinds of tests for a patient.

This system will reduce patient and doctor's work. Patient doesn't have to visit hospital to get appointment. It will save a lot of time.

Functions:

void Login (String username, String pin);

This function verifies the username and pin entered by the user. And if the username and pin does not match with the username and pin stored in the database then he will have to re-enter the username and pin till he enters the correct username and password.

void getAppointment (String name, Date date);

In this function the user will request for appointment. He will enter date for that appointment. Then the appointment will be reserved for the specific date.

void registerPatient (String name, String pin, String mobile);

This function allows user to register himself in the system. He will have to enter name, pin and mobile. Then he will get registered in the system.

Black box Testing:

Worst BVA:

Function 1:

Total cases = 5 ^ 1 = 5

Min: 0

Min+1: 1

Normal: 50000

Max -1: 99998

Max: 99999

Case	pin	Expected output
1	0	Invalid
2	24324	Valid
3	34643	Valid
4	59786	Valid
5	99999	Invalid

Function 2:

Total cases = 5 ^ 1 = 5

Min: a

Min + 1: b

Normal: h

Max-1: y

Max: z

Case	name	Expected output
1	Hamza	Valid
2	Fahad	Valid
3	a	Invalid
4	Ali	Valid
5	z	Invalid

Function 3:

Input values of name:

Min: a

Min + 1: b

Normal: h

Max-1: y

Max: z

Input values of pin:

Min: 0

Min+1: 1

Normal: 50000

Max -1: 99998

Max: 99999

Input values of email:

Min: a

Min + 1: b

Normal: h

Max-1: y

Max: z

Case	name	pin	email	Expected output
1	Ali	00000	Bse301053@cust.pk	Invalid
2	Ali	00001	Bse301053@cust.pk	Valid
3	Ali	00002	Bse301053@cust.pk	Valid
4	Ali	00003	Bse301053@cust.pk	Valid
5	Ali	00005	Bse301053@cust.pk	Valid
6	Ali	00006	Bse301053@cust.pk	Valid
7	Ali	00007	Bse301053@cust.pk	Valid

8	Ali	00008	Bse301053@cust.pk	Valid
9	Ali	00009	Bse301053@cust.pk	Valid
10	Ali	000010	Bse301053@cust.pk	Valid
11	Ali	00011	Bse301053@cust.pk	Valid
12	Ali	00012	Bse301053@cust.pk	Valid
13	Ali	00013	Bse301053@cust.pk	Valid
14	Ali	00014	Bse301053@cust.pk	Valid
15	Ali	00015	Bse301053@cust.pk	Valid
16	Ali	00016	Bse301053@cust.pk	Valid
17	Ali	00017	Bse301053@cust.pk	Valid
18	Ali	00018	Bse301053@cust.pk	Valid
19	Ali	00019	Bse301053@cust.pk	Valid
20	Ali	00020	Bse301053@cust.pk	Valid
21	Ali	00021	Bse301053@cust.pk	Valid
22	Ali	00022	Bse301053@cust.pk	Valid
23	Ali	00023	Bse301053@cust.pk	Valid
24	Ali	00024	Bse301053@cust.pk	Valid
25	Ali	00025	Bse301053@cust.pk	Valid
26	Hamza	00026	Bse181053@cust.pk	Valid
27	Hamza	00027	Bse181053@cust.pk	Valid
28	Hamza	00028	Bse181053@cust.pk	Valid
29	Hamza	00029	Bse181053@cust.pk	Valid
30	Hamza	00030	Bse181053@cust.pk	Valid
31	Hamza	00031	Bse181053@cust.pk	Valid
32	Hamza	00032	Bse181053@cust.pk	Valid
33	Hamza	12300	Bse181053@cust.pk	Valid
34	Hamza	12311	Bse181053@cust.pk	Valid
35	Hamza	12323	Bse181053@cust.pk	Valid
36	Hamza	12367	Bse181053@cust.pk	Valid
37	Hamza	12343	Bse181053@cust.pk	Valid
38	Hamza	12327	Bse181053@cust.pk	Valid
39	Hamza	34560	Bse181053@cust.pk	Valid
40	Hamza	34561	Bse181053@cust.pk	Valid
41	Hamza	34562	Bse181053@cust.pk	Valid
42	Hamza	34563	Bse181053@cust.pk	Valid
43	Hamza	34564	Bse181053@cust.pk	Valid
44	Hamza	34565	Bse181053@cust.pk	Valid
45	Hamza	34566	Bse181053@cust.pk	Valid
46	Hamza	34567	Bse181053@cust.pk	Valid
47	Hamza	34568	Bse181053@cust.pk	Valid
48	Hamza	34569	Bse181053@cust.pk	Valid
49	Hamza	34510	Bse181053@cust.pk	Valid
50	Hamza	67800	Bse181053@cust.pk	Valid
51	Fahad	67890	Bse181056@cust.pk	Valid
52	Fahad	67891	Bse181056@cust.pk	Valid
53	Fahad	67892	Bse181056@cust.pk	Valid
54	Fahad	67893	Bse181056@cust.pk	Valid
55	Fahad	67894	Bse181056@cust.pk	Valid
56	Fahad	67895	Bse181056@cust.pk	Valid
57	Fahad	67896	Bse181056@cust.pk	Valid

58	Fahad	67897	Bse181056@cust.pk	Valid
59	Fahad	67897	Bse181056@cust.pk	Valid
60	Fahad	99999	Bse181056@cust.pk	Invalid

Strong Robust Equivalence:

Function 1: void Login (String username, String pin);

Pin>=0 and Pin<=99999

Test cases are:

(-1,55555,100000)

Function 2: void getAppointment (String name, Date date);

NameLength>=5 and <=10

Test cases are:

(Ham, Shahzad, RajaFarhanaa)

Function 3: void registerPatient (String name, String pin, String mobile);

1) Pin>=0 and Pin<=99999

Test cases are:

(-1,55555,100000)

2) NameLength>=5 and <=10

Test cases are:

(Ham, Shahzad, RajaFarhanaa)

3) Mobile number >= 0 and <= 11

Test cases are:

(-1,6,12)

case	Pin	Name	Mobile	Output
1	-1	Ham	-1	Invalid
2	-1	Shahzad	6	Invalid
3	-1	RajaFarhanaa	12	Invalid
4	-1	RajaFarhanaa	-1	Invalid
5	55555	Faha	6	Invalid
6	55555	Ham	12	Invalid

7	55555	RajaFarhanaa	-1	Invalid
8	55555	Fahad	6	valid
9	99999	Hamza	12	valid
10	100000	RajaFarhan	1	valid
11	100000	Faha	6	Invalid
12	100000	Hamza	12	Invalid

Function 1 Requirements:

Function 1: void Login (String username, String pin);

If username is less than 5 words and pin is less than 9 words then user will not get login in the system.

If username is greater 5 words and pin is equal to 8 words then user will get login into the system.

If username is greater than 5 words and less than 9 words then user will get login into the system.

If pin is greater than 5 words and less than 9 words then user will get login into the system.

If username is less than 9 words and pin is greater than 8 words then user will not get login into the system.

Causes/Effects:

C1: username<5

C2: pin<9

C3: username>5

C4: pin=8

C5: username>5

C6: pin>5

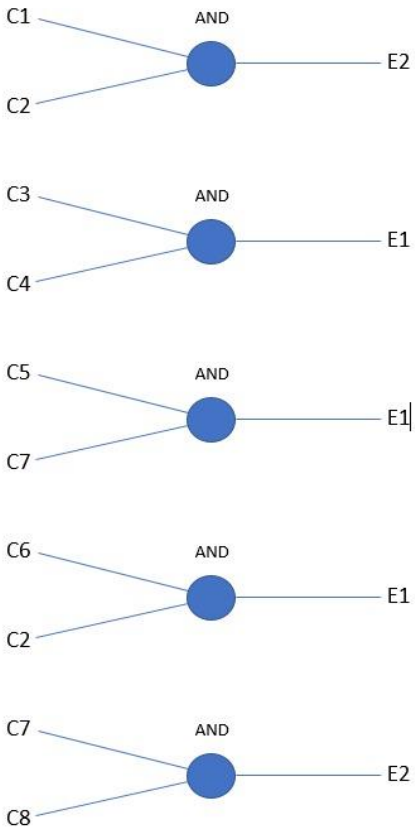
C7: username<9

C8: pin>8

E1: Login

E2: not Login

Graph



Decision Table

Conditions/Effects	1	2	3	4	5
C1: username<5	1	0	0	0	0
C2: pin<9	1	0	0	1	0
C3: username>5	0	1	1	0	0
C4: pin=8	0	1	0	0	0
C5: username>5	0	0	1	0	0
C6: pin>5	0	0	0	1	0
C7: username<9	0	0	0	0	1
C8: pin>8	0	0	0	0	1
E1: Login	0	1	1	1	0
E2: Not Login	1	0	0	0	1

EQP Test Cases:

Valid Class if username Length= {6,7,8}

Valid Class if pin= {6,7,8}

Invalid Class if username= {9,10, 11, …….}

Invalid Class if Pin= {9, 10, 11, ……}

Reason to use EQP:

We used EQP because in this technique, input data units are divided into equivalent partitions that can be used to derive test cases which reduces time required for testing because of small number of test cases.

Function 2 Requirements:

Function 2: void getAppointment (String name, Date date);

If name is not valid then patient will not get appointment.

If name is valid then patient will get appointment.

Causes/Effects:

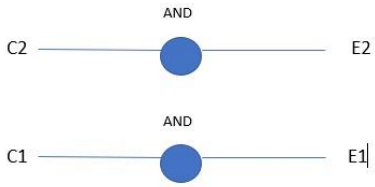
C1: username= valid

C2: username is not valid

E1: get Appointment

E2: not get Appointment

Graph



Decision Table

Conditions/Effects	0 1
C1: username= valid	1 0
C2: username is not valid	0 1
E1: get Appointment	1 0
E2: not get Appointment	0 1

EQP Test Cases:

Valid if username= {Hamza, Fahad, Farhan, …….}

Invalid if username= {xyz, abc, hys, ……}

Function 3 Requirements:

Function 3: void registerPatient (String name, String pin, String mobile);

If name is less than 5 words and pin is less than 9 words then patient will not get registered.

If name is greater than 5 words and pin is less than 9 words then patient will get registered.

If name is greater than 5 words and pin is less than 5 words then patient not will get registered.

If name is greater than 8 words and pin is less than 9 words then patient will get not registered.

If name is greater than 8 words and pin is greater than 9 words then patient will not get registered.

If name is greater than 5 words and less than 8 words then patient will get registered.

If pin is greater than 5 words and less than 9 words then patient will get registered.

If name is greater than 5 words and pin is greater than 5 words then patient will get registered.

Causes/Effects:

C1: name<5

C2: name>5

C3: name<8

C4: name>8

C5: pin<9

C6: pin>9

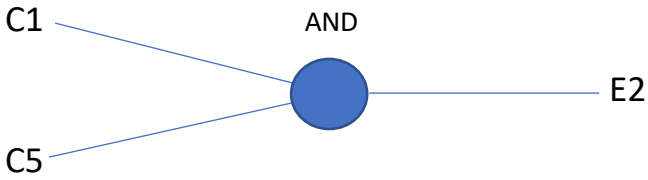
C7: pin>5

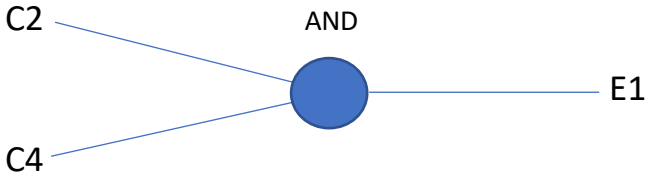
C8: pin<5

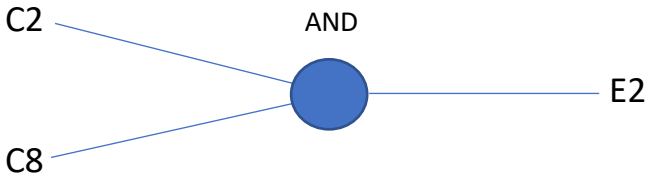
E1: get Registered

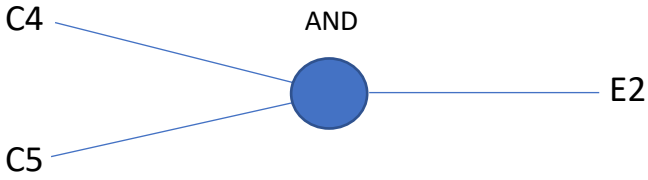
E2: not get Registered

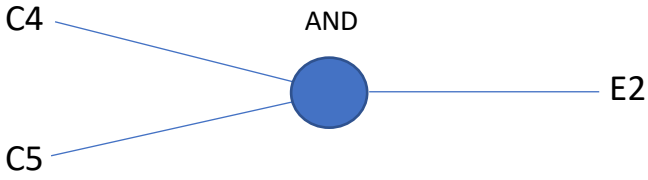
Graph

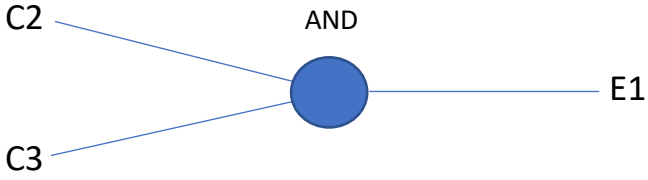

C1
AND
E2
C5

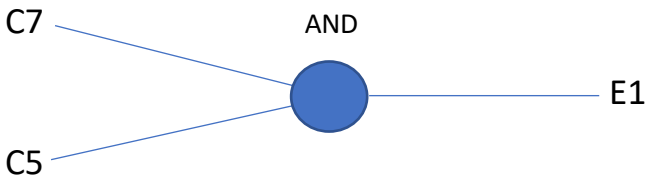

C2
AND
E1
C4

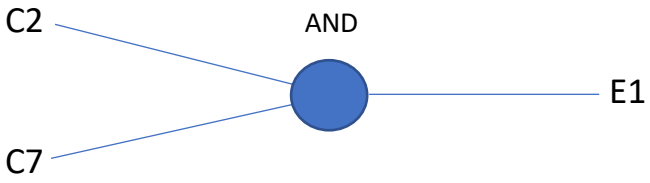

C2
AND
E2
C8


C4
AND
E2
C5


C4
AND
E2
C5


C2
AND
E1
C3





Decision Table

Conditions/Effects	0	1	2	4	5	6	7	8
C1: name<5	1	0	0	0	0	0	0	0
C2: name>5	0	1	1	0	0	1	0	1
C3: name<8	0	0	0	0	0	1	0	0
C4: name>8	0	0	0	1	1	0	0	0
C5: pin<9	1	1	0	1	0	0	1	0
C6: pin>9	0	0	0	0	1	0	0	0
C7: pin>5	0	0	0	0	0	0	1	1
C8: pin<5	0	0	1	0	0	0	0	0
E1: get Registered	0	1	0	0	0	1	1	1
E2: not get Registered	1	0	1	1	1	0	0	0

EQP Test Cases:

Valid if username Length= {6, 7, 8}

Valid if pin Length= {6,7,8}

Invalid if username Length= {9, 10, 11, ……..}

Invalid if pin Length= {9, 10, 11, …….}