Medical Center

Chapter 1
Introduction

PURPOSE:

Creating Software Helps Us To Facilitate The Medical Process And Solve Some Of The Problems Facing Doctor , Assistant And Patients. Solutions To These Problems Include Adding Accounts To Facilitate The Collection Of Data About Patients And Increase The Security Level Of The System.

SCOPE:

This System Helps The Doctor , Assistant And Patients To Facilitate The Medical Process.

OBJECTIVES:

Making Some Modifications To The Old System By Adding New Features:

-Create An Account For Each Patient To identify Him And Store His Data.

-Establishing An Advanced Level Of Security To Determine The Authority Of Both The Employer And Patient While Using The System.

-Providing An Easy-To-Use print Reports Command To Facilitate The Task Of Obtaining Some Information From The System.

-Providing An Easy-To-Use Interface And Icons Whose Functions Are Easy To Understand.

DEFINITION:

Medical Centers Are Community-Based, Patient-Oriented Organizations That Provide Comprehensive, Culturally Competent, High Quality health Care Services.

TERMINOLOGY:

1.Doctor: Diagnose And Treat A Variety of Diseases And Injuries. Annual Health Examination. Conducts Many Tests, Analyzes And Diagnostic Images To Provide Information About The Patient’s

Condition. Analyzes Reports And Results From The Tests And Works To Diagnose The Condition.

2.Assistant: The Medical Assistant Performs Several Administrative Tasks, Such As Answering The Telephone, Receiving Patients, Updating Patient Information And Recording It In Medical Records, Filling Out Insurance Forms, Handling Approvals, Organizing Appointments, Arranging Center Admissions And Laboratory Tests, And Handling Accounting And Bookkeeping.

3.Patient: The Patient Is The Main person In The Center Because Most Of The Basic Operations Are Based On The Data He Provides. He Is The One Who Books An Appointment To Visit The Doctor To Diagnose The Disease He Is Suffering From.

ABBREVIATIONS:

API: Application Programming Interface , They Are Tools That Enable Programs To Communicate And

Perform A Series Of Tasks. In The World Of Banking,

These Interfaces Provide Secure Access To Financial Services Via External Platforms, Helping Companies Build Innovative Banking-Focused Products.

Chapter 2
Feasibility study

OPERATIONAL FEASIBILITY:

The Clint Were Facing Some Problems In His System Like:

1-Difficult In Collecting All Information About Each Patient Manually Without Any Loss Or Error In Registration, So We Are Going To Working On Making An Account For Each Patient To Be Easy To Collect Data Without Mistakes.

2-Difficult In Scheduling Doctors' Appointments If There Is More Than One Doctor At The Same Time, So We Will Make A List To Register Appointments.

3-Difficulty Communicating With Patients In An Emergency Situation Such As The Absence Of The Doctor, So We Will Make It Easy By Sending Notifications To Patients.

4- Difficulty In Organizing The Patient's Appointment Schedule, So We Will Make The Registration Electronic From The Patient.

TECHNICAL FEASIBILITY:

To Make Client's Requirement And Solve The Problems Facing Him We Will Build One Application On PC To The Client Which Solves All His Problems.

RISKS THAT MIGHT BE FACING US IS:

1-Building The System May Require Longer Time Than Agreed Time And For Can Specify Time More That We Need With Customer.

2-Application Size May Be Bigger That The Space That The Server Can Handle It So We Will Try To Make It With Suitable Space.

3-If User Not Expert Enough He May Face Problems With Using That Application So We Will Make It Simple As Possible To Avoid That And We Can Give User Some Guides To Help Him With Using It.

4-Misunderstanding Of The Client's Requirements May Occur. To Avoid This, We Will Present The Completed Parts Of The Project To The Client Periodically.

ECONOMIC FEASIBILITY:

TANGIBLE BENEFITS:

1-Increased Profit Due To The System's Ability To Handle A Large Number Of Patient In An Organized Manner And I Tracking Patient Periodically, Thus Increasing The Attraction Of Larger Number Of Patients, Which Lead To Increased Profits.

2-The Application Increased Speed Of Activity.

3-It Reduce Rate Of Human Mistakes So That It Reduces The Money The Was To Be paid To Solve This Mistakes.

4-Cost Saving By Replacing Manual Processes With An Application, The Medical Center Can Potentially Reduce Costs Associated With Paper-Based System.

INTANGIBLE FENEFITS:

1-Improve Decision-Making: Access To Accurate And Timely Data Through The Application Can Support Informed Decision-Making By Assistant

And Doctor, Leading To Improved Medical Outcome And Operational Efficiency.

2-Enhanced Satisfaction: By Allowing To System To Make A Report Contains Patient Information.

3-Enhanced Collaboration: An Application Can Facilitate Collaboration Between Which Use It.

4-Competitive Advandage: Implementing An Application To Manage Medical Center Data Can Differentiate The Center From Competitors.

5-Future-Proofing: Investing In An Application Demonstrates A Forward-Thinking Approach And Adaptability To Technological Advancements, Positioning The Medical Center For Future.

TANGIBLE COSTS:

1-Development And Implementation Costs: Developing A Custom Application Or Customizing An Existing Solution Involves Costs Associated With Hiring Developers.

2-Maintenance And Support Costs: Ongoing Maintenance, Updates And Technical Support For The Application Require Financial Resources To Ensure Its Smooth Operation And Address Any Issues That May Arise.

3-Training Costs: Training To User To Use The application Effectively Requires Time, Effort And Potentially External Training Resources.

INTANGIBLE COSTS:

1-Time And Effort: Developing And Implementing An Application Requires A Significant Investment Of Time And Effort.

2-Change Management: Introducing A New Application May Require Change Management Efforts To Ensure Smooth Adoption.

Chapter 3
Proposal System Analysis And Design

INREVIEW:

-Interview With Doctor

***Open Ended**

١- ما دورك فى ادارة المجمع الطبى؟

تقديم الرعاية الطبية،وتحديد سبب المرض وتقديم العلاج المناسب مع مراعاة التاريخ المرضى للمريض.

٢- ما اكبر الصعوبات التى تواجهك لاكمال عملك؟

١.صعوبة جدولة المواعيد واهدار الكثير من الوقت فى تنظيم الوقت مما يؤدى الى انتظار المرضى الكثير من الوقت.

٢.فقدان اوتلف السجلات الورقية مما يؤدى الى صعوبة فى استرجاع بيانات المريض بسرعة،ومشاكل فى متابعة التاريخ المرضى.

٣.عدم التزام المرضى بالمواعيد المحددة يؤدى الى حالات غياب كثيرة واهدار الكثير من وقت الطبيب.

٣- ما البيانات الاساسية التى تحتاجها من ملف المريض؟

١.العمر ٢.التاريخ المرضى بالكامل

٣.هل يعانى من اى امراض مزمنة

٤.هل اجرى عمليات جراحية من قبل

٥.هل لديه اى حساسية

٦.هل يأخد علاج بصورة مستمرة

٤- ما هى الخدمات التى تريد اضافتها فى النظام الجديد؟

١.تقديم جدول مواعيد اليوم بالكامل

٢.بيانات شاملة عن المريض قبل الزيارة

٣.خصوصية بيانات المريض و صلاحية الوصول للبيانات

٤.القدرة على التعديل على البيانات

***Closed**

١- هل يحتوى المركز على نظام بالفعل؟

-نعم - لا

٢- هل تتلقى المعلومات عن المريض قبل الزيارة؟

-نعم -لا

٣-هل توجد صعوبة فى التعامل مع النظام الحالى؟

-نعم -لا

٤- هل تحتاج لتحديث بيانات المريض بشكل دورى؟

-نعم -لا

Interview With assistant:

***Open Ended**

١- ما دورك فى النظام؟

١.جدولة مواعيد الاطباء

٢.تسهيل وتنظيم العملية الطبية بين المريض والطبيب

٣.تنظيم دخول المرضى ٤.ادارة الحسابات والدفع

٥.التأكد من كفاءة الطبيب عن طريق التحدث مع المريض

٢- ما الصعوبات التى تواجهك؟

١.فى حالة وجود طوارئ يؤثر علىوقت الحجوزات

٢.وجود فى بعض الاحيان تعارضات وعدم دقة فى الحجز بسبب بعض العوامل الخارجية الغير متوقعة

٣.عدم وجود تذكيرات تلقائية للمرضى يؤدى الى كثرة غياب المرضى او تاخيرهم

٤.متابعة المدفوعات يدويا صعبة وقد تؤدى الى اخطاء مالية

٥.تأمين البيانات يكون اصعب لأن الملفات الورقية ممكن تضيع او تتلف بسرعة

***Closed**

١- هل توجد صعوبة فى جدولة المواعيد للاطباء؟

-نعم -لا

٢- هل تجد صعوبة فى استرجاع بيانات المرضى ؟

-نعم -لا

٣- هل تحدث اخطاء متكررة اثناء تسجيل المرضى؟

-نعم -لا

٤- هل يتم ارشفة الملفات بطريق تسهل الوصول اليها؟

-نعم -لا

٥- هل يتم تسجيل المواعيد يدويا؟

-نعم -لا

Interview With Patient

*Open Ended

١- ما المشاكل التى تواجهك؟

١.ضياع الوقت اثناء الانتظار وفى بعض الاوقات اتفاجئ بعدم حضور الطبيب

٢.عدم وجود تذكير بالمواعيد،وتأخير فى تسجيل البيانات و وجود ضرورة فى تكرار المعلومات اكثر من مرة

٣.التكاليف مش واضحة،والتفاجئ بالسعر بعد الكشف ووجود حالة ان يكون السعر واضح قبل بداية الكشف

٤.عدم وجود خيارات دفع الكترونى،والحاجة الى دفع كاش

*Closed

١- هل سبق استخدام تطبيق حجز من قبل؟

-نعم -لا

٢- هل وجود تطبيق او موقع الكترونى لمتابعة حالتك الصحية؟

-نعم -لا

٣- هل وجود نظام سوف يساعدك فى توفير وقتك؟

-نعم -لا

٤- هل تفضل الدفع الالكترونى عن الكاش؟

-نعم -لا

OBSERVATION:

1-The serve space which is used in the system is low.

2-The desire increase the security and privacy of data.

3-Desire to develop at low cost.

REQUIREMENTS:

***Functional**

1-Having a program with an easy-to-use interface.

2-Organize each doctor's appointment schedule.

3-Easy access to and modification of information.

4-Continuous communication with the patient in the presence of an emergency that affects the reservation.

5-Organize your booking schedule.

6-Sending some alerts to patients in case the patient does not attend.

7-Possibility of paying electronically.

8-Recording information in a correct and orderly manner.

9-Continuous modification of information.

10-Information privacy.

***Non-Functional:**

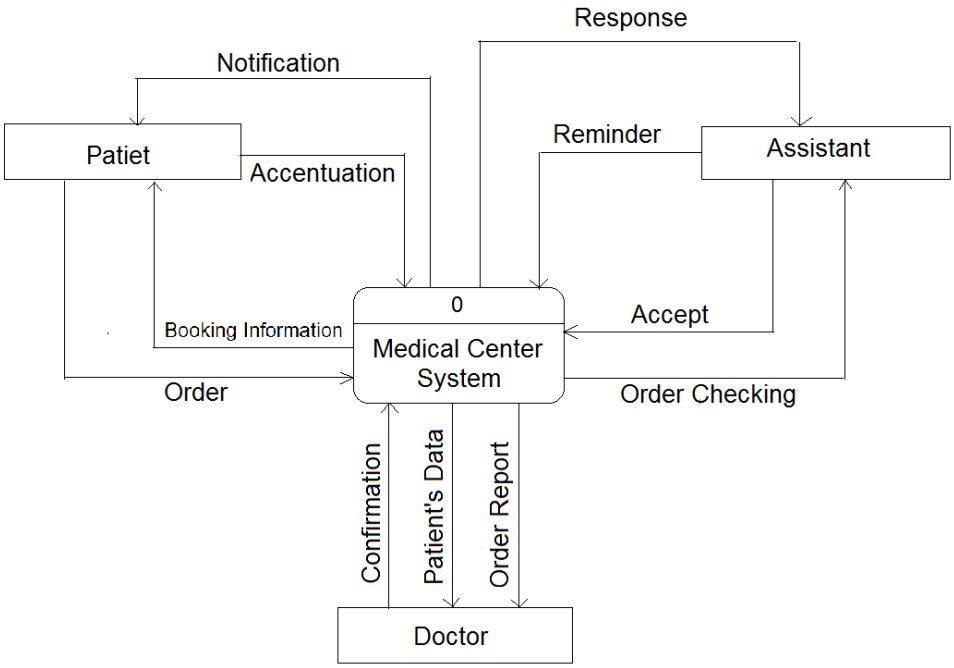
1-Performance:The system should response to user in 1 or 2 seconds to ensure a smooth user experience.

2-Scalability:The system should be able to carry a large number of data without any degradation in performance and the infrastructure.

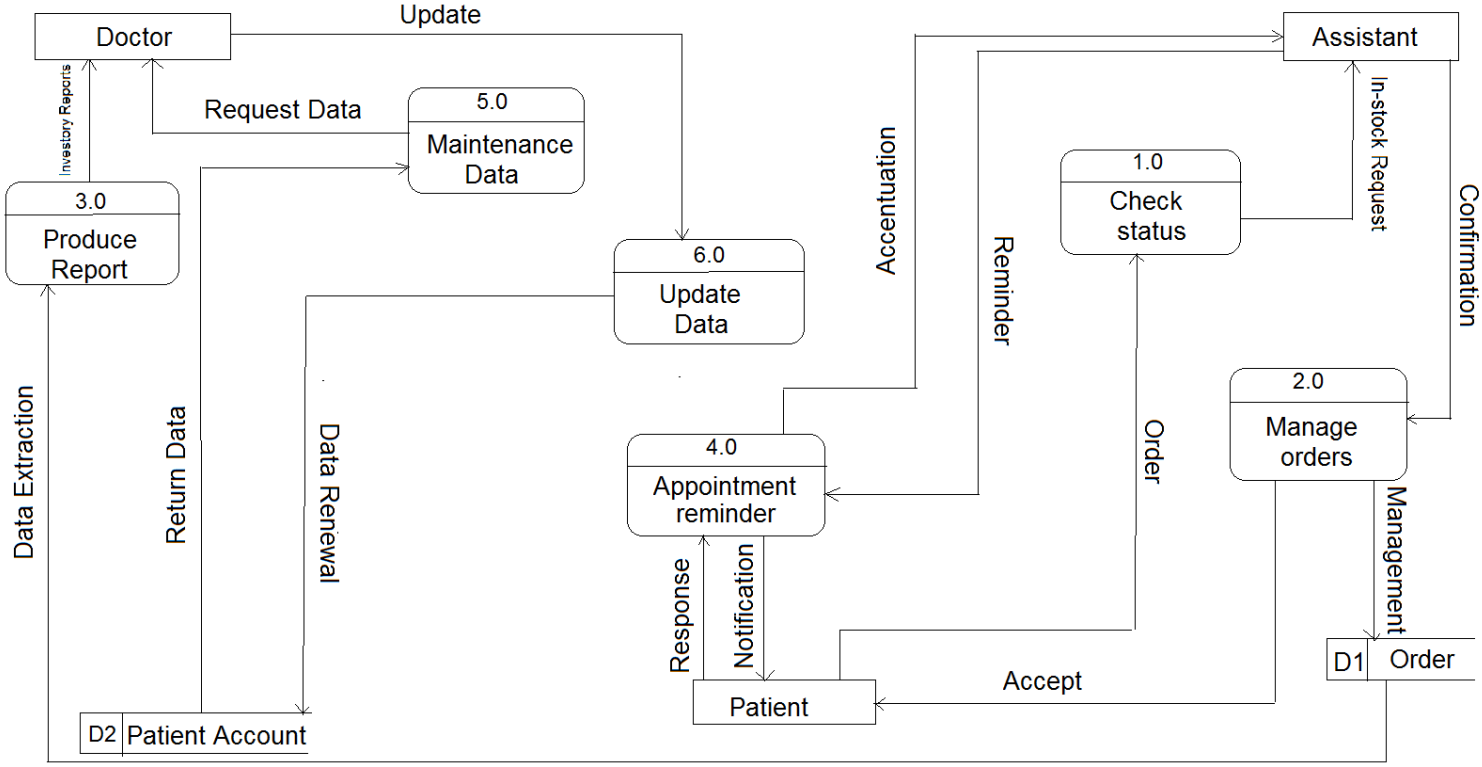
3-Security:System should have a high level of security .

4-Usability:Te user interface should be intuitive and easy to navigate,allowing users to quickly find information.

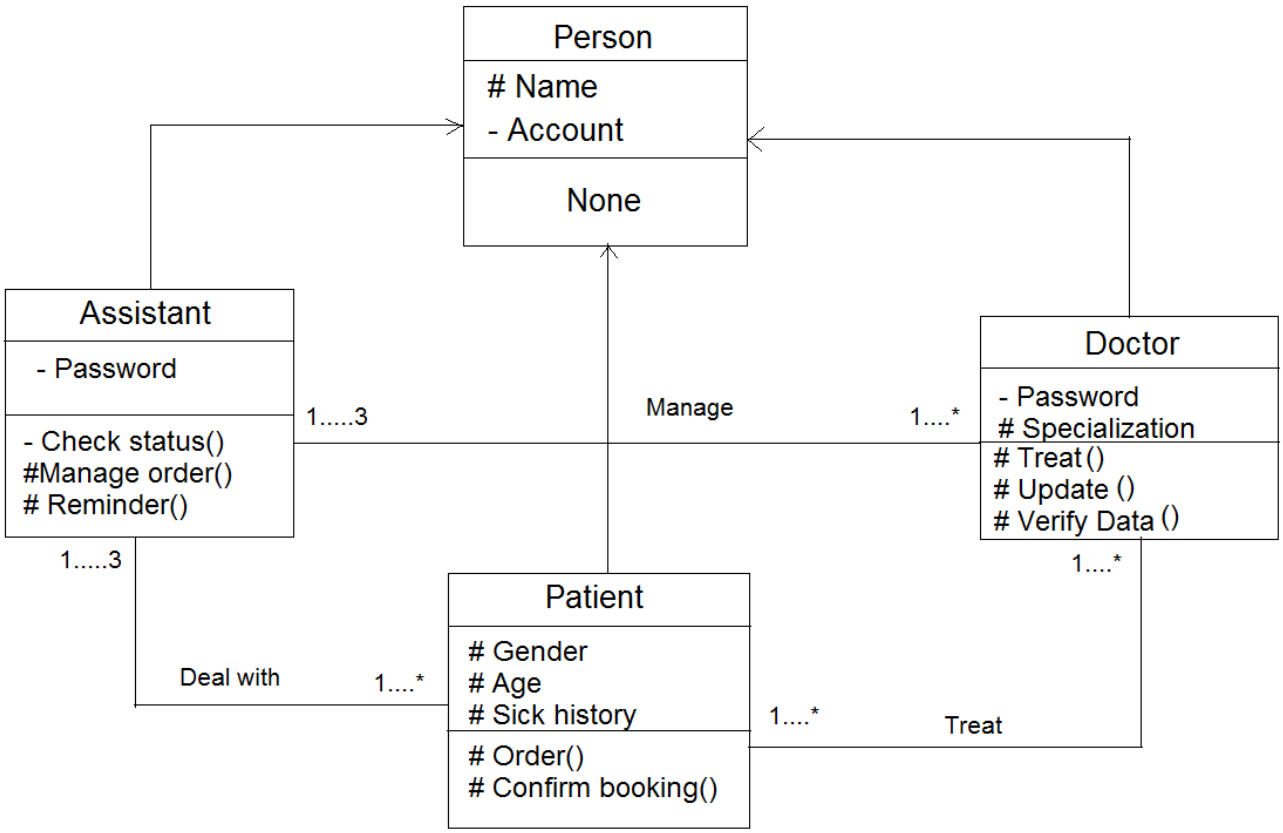
COTEXT DIAGRAM



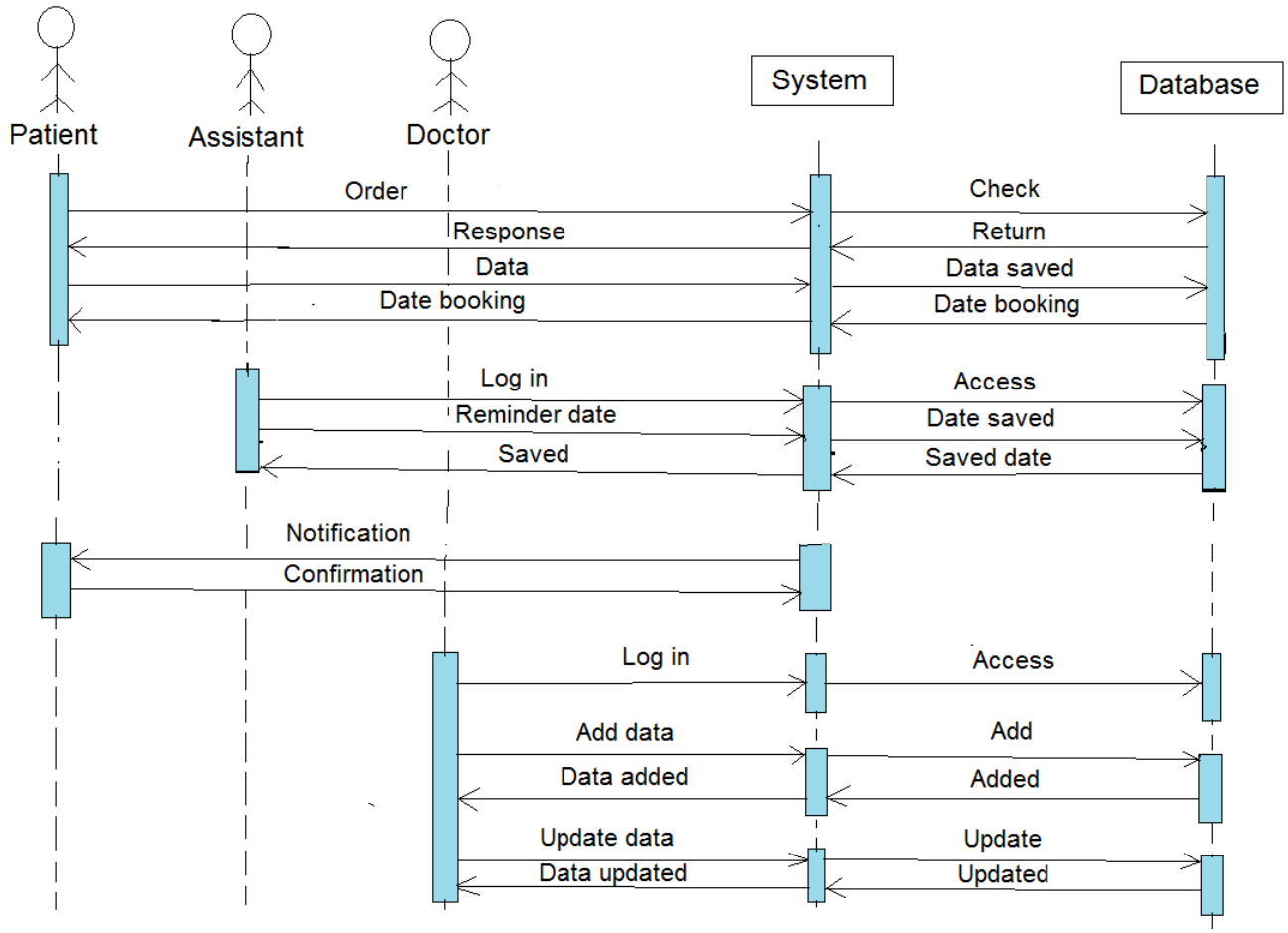
DFD LEVEL 0


Doctor
Update
Assistant
Request Data
5.0
Maintenance Data
Investory Reports
3.0
Produce Report
6.0
Update Data
Accentuation
Reminder
1.0
Check status
In-stock Request
Confirmation
2.0
Manage orders
Order
Management
4.0
Appointment reminder
Response
Notification
Data Extraction
Return Data
Data Renewal
D1
Order
Accept
Patient
D2
Patient Account

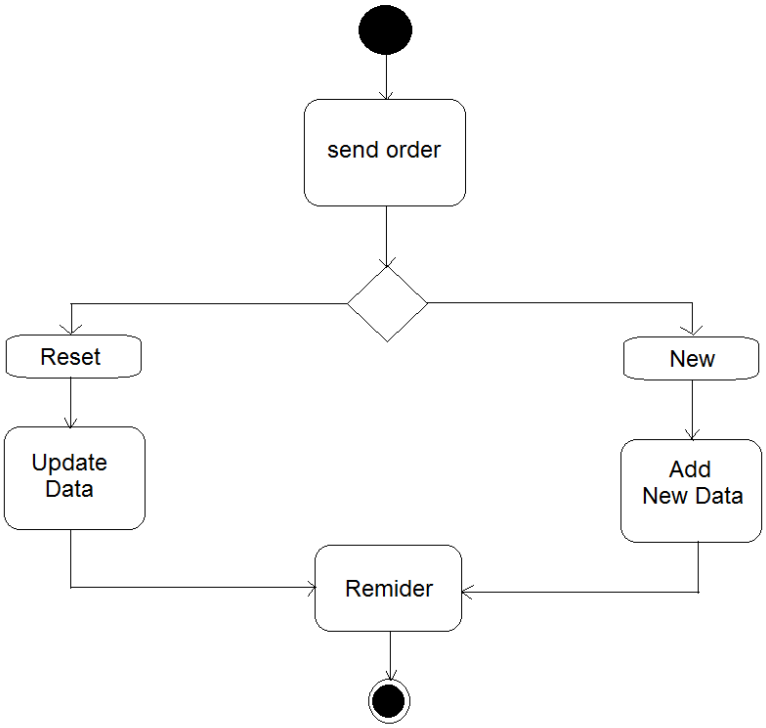
CLASS DIAGRAM

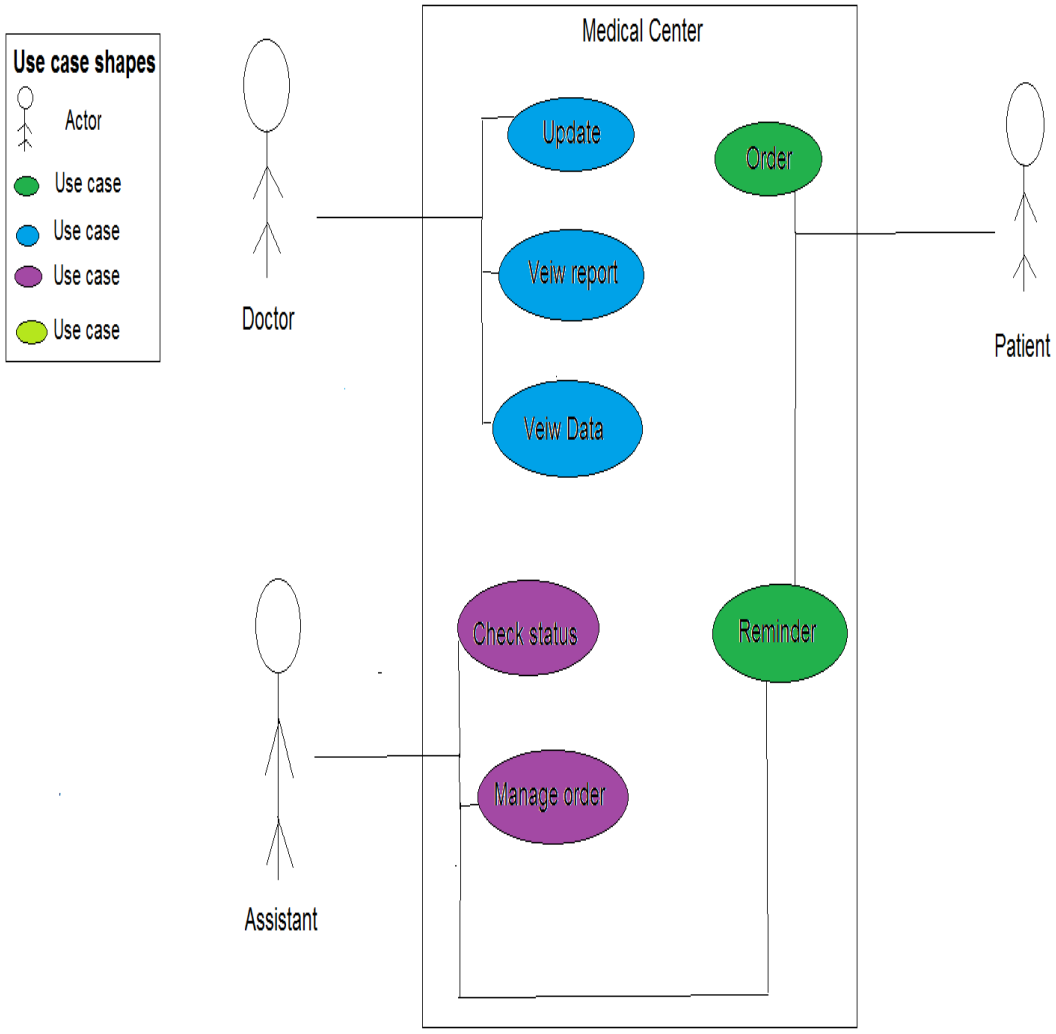


Sequential DIAGRAM



ACTIVITY DIAGRAM




Use case shapes
Actor
Use case
Use case
Use case
Use case
Medical Center
Update
Order
Veiw report
Veiw Data
Check status
Reminder
Manage order
Doctor
Patient
Assistant

USE CASE

FLOW OF USE CASE:

ACTORS:

-Doctor

-Assistant

-Patient

USE CASE:

-Order

-Check status

-Manage order

-Reminder

-View report

-View data

-Update

Flow of event #1:

>> Use Case: Order

>> Actors : Patient

>> Description : The patient submits a request to book an appointment with the doctor, specifies the specialty, and sends his medical history to be recorded for the doctor to review during the examination process.

Flow of event #2:

>> Use Case: Check status

>> Actors : Assistant

>> Description : The assistant determines the booking date based on the schedule.

Flow of event #3:

>> Use Case : Manage order

>> Actors : Assistant

>> Description : The assistant records the appointment in the appointment schedule and sends the order confirmation to the patient.

Flow of event #4:

>> Use Case : Remider

>> Actors : Assistant , Patient

>> Description : The assistant specifies a specific date before the appointment time to send a reminder to the patient, and the patient confirms the appointment.

Flow of event #5:

>> Use Case : View report

>> Actors : Doctor

>> Description : The doctor checks his appointment schedule daily to see how many checkups he has.

Flow of event #6:

>> Use case : View data

>> Actors : Doctor

>> Description : The doctor reviews the patient's medical history to know the disease, how to

diagnose it, and prescribe the appropriate medication.

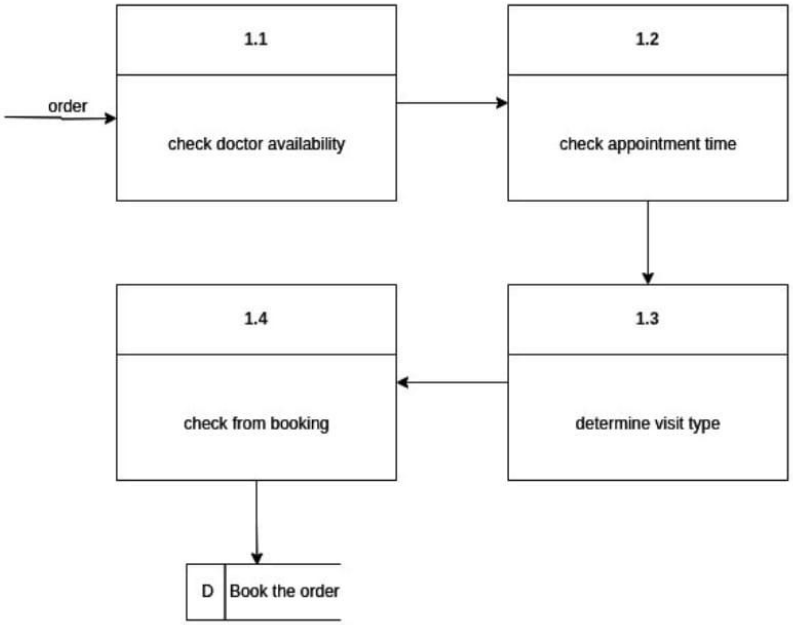
Flow of event #7:

>> Use Case : Update data

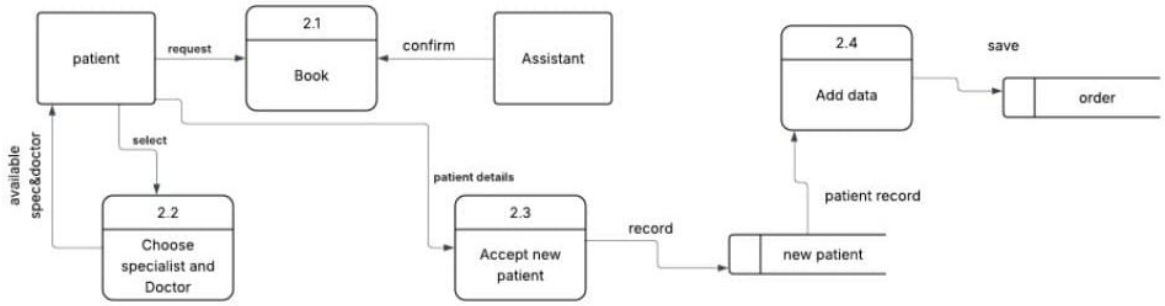
>> Actors : Doctor

>> Description : After the doctor completes the examination, he modifies the recorded data to be ready when the patient comes again for a re-examination.

LEVEL1:PROCESS.1(Check status)

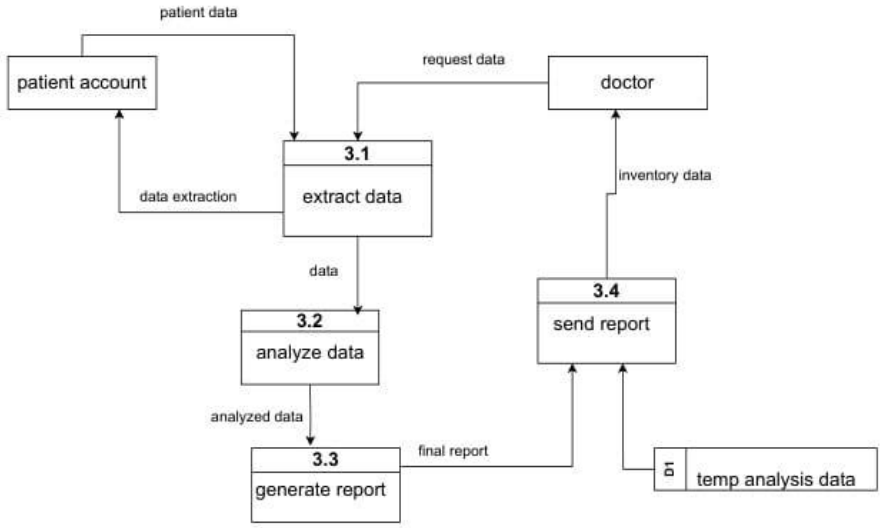


LEVEL1:PROCESS.2(Manage orders)

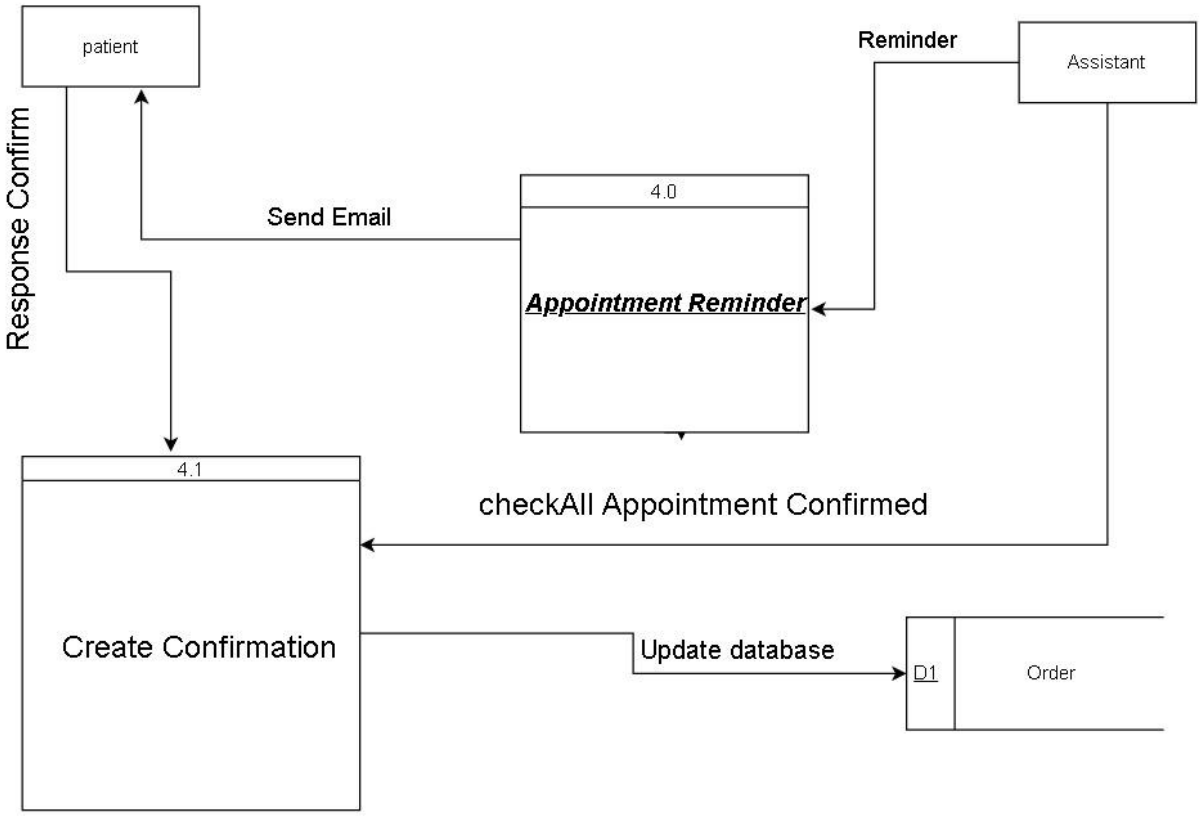


LEVEL1:PROCESS.3(Produce report)

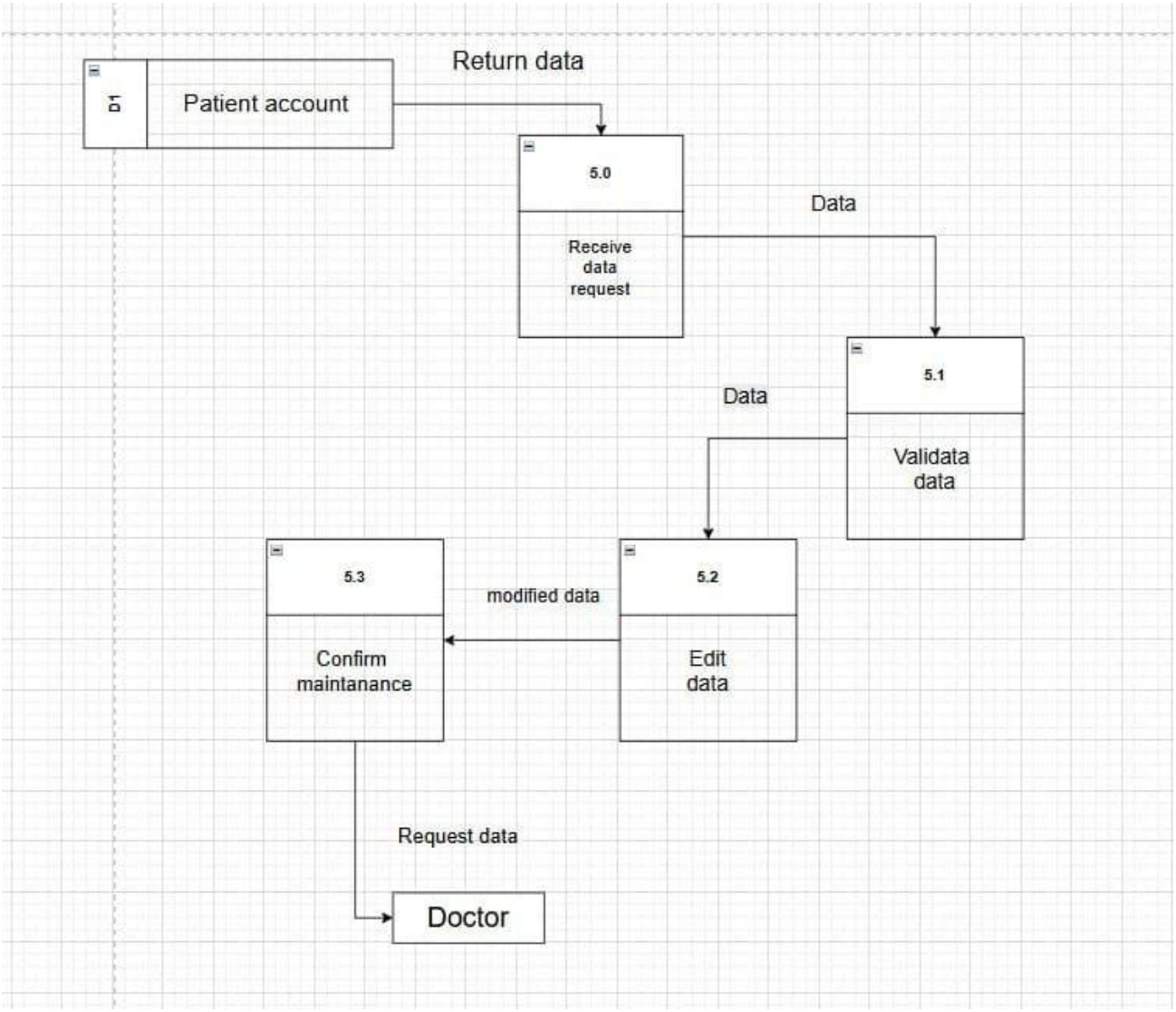
DFD level 1 for 3.0 produce report:



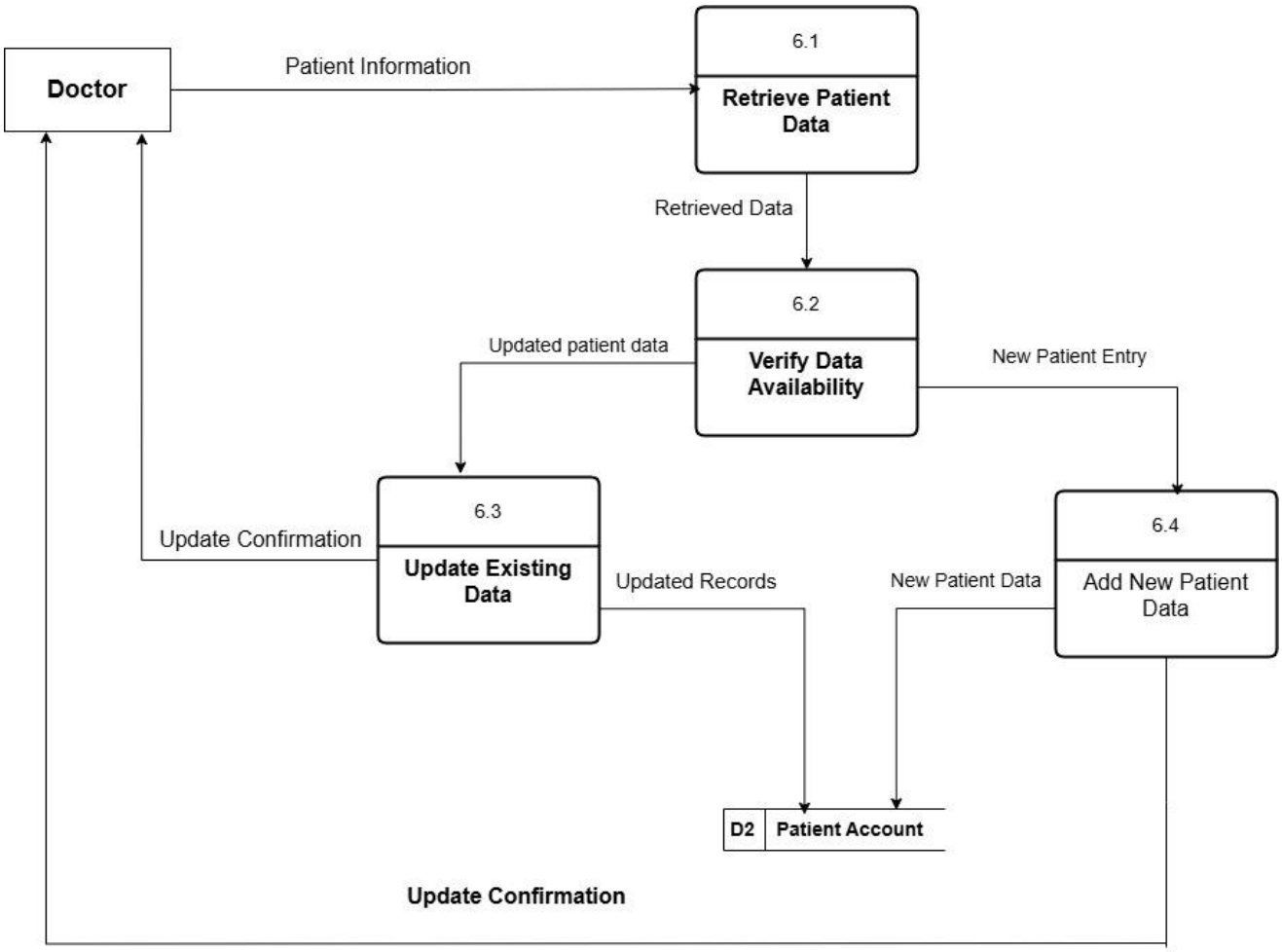
LEVEL1:PROCESS.4(Appointment reminder)



LEVEL1:PROCESS.5(Maintenance data)



LEVEL1:PROCESS.6(Update data)



Logical modeling for process1:

```
DO
READ THE ORDER
if doctor is available

    if appointment time is available
        make booking
    else
        if have re-examination
            make booking
        else
            send message all time is booked

else
    send message Doctor is not available
on this day
```

Decision Table For Check Status

	Doctors availability	C	C	C	C	I	I	I	I
	appointment time	R	R	F	F	R	R	F	F
Condition	Medical re-examination	True	False	True	False	True	False	True	False
	Agree	x	x	x					
Action	Not Agree				x	x	x	x	x

Decision Table For Check Status

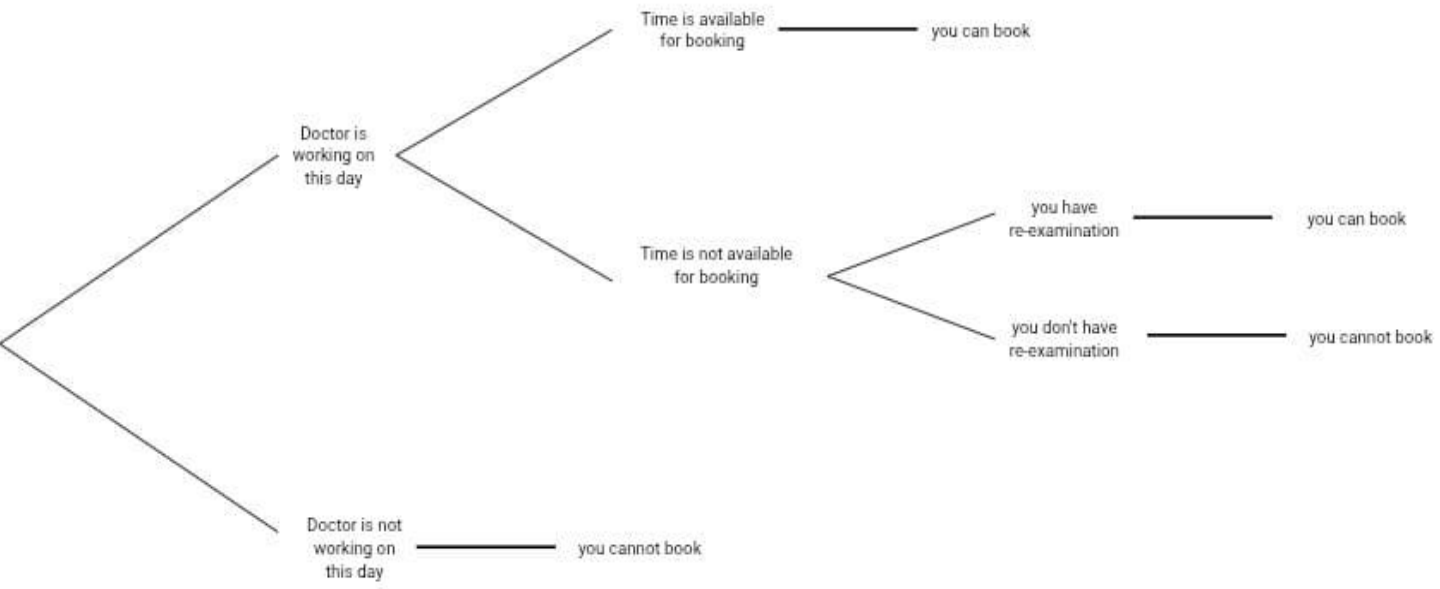
	Time for Doctors	C	C	C	I				
	Time for Patients	R	F	F	----				
Condition	Medical re-examination	----	True	False	----				
	Agree	x	x						
Action	Not Agree			x	x				

Doctors availability => C

Doctors unavailability => I

appointment time is availability => R

appointment time is unavailability => F


Doctor is working on this day
Time is available for booking
you can book
Time is not available for booking
you have re-examination
you can book
you don't have re-examination
you cannot book
Doctor is not working on this day
you cannot book

Logical modeling for process2:

Process 2.0: Manage order.

Do

Read New patient confirmation

Find matching patient record

If it exists

Add data to order

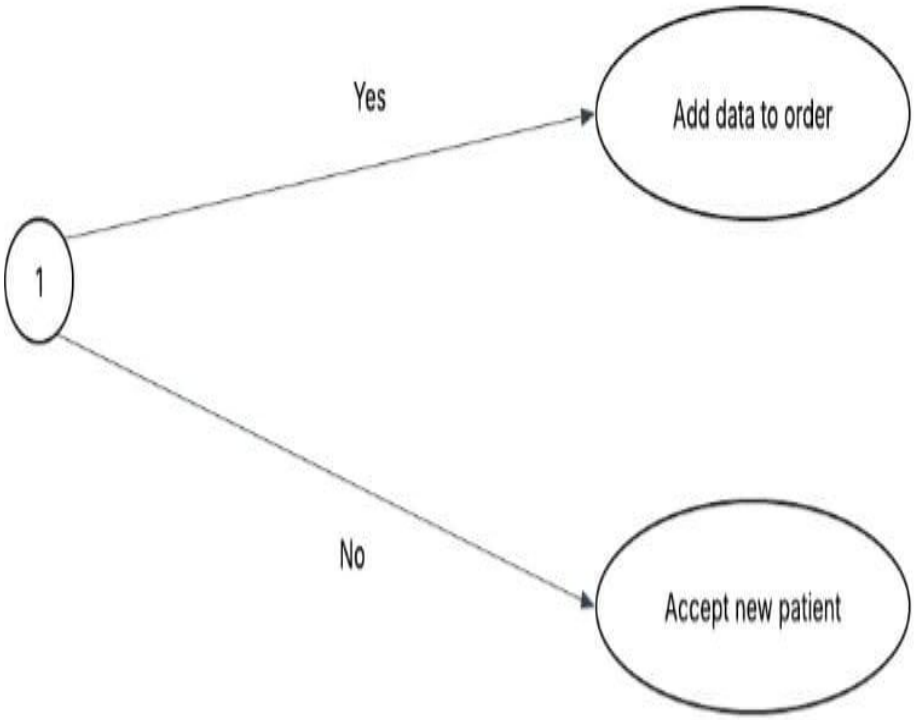
Else

Accept new patient

UNTIL End-of - file

	Conditions / Courses of Actions	Rules	
		1	2
Condition stubs	New patient confirmation received	Y	Y
	Matching patient record exists	Y	N
Action stubs	Add data to order	✗	
	Accept new patient		✗

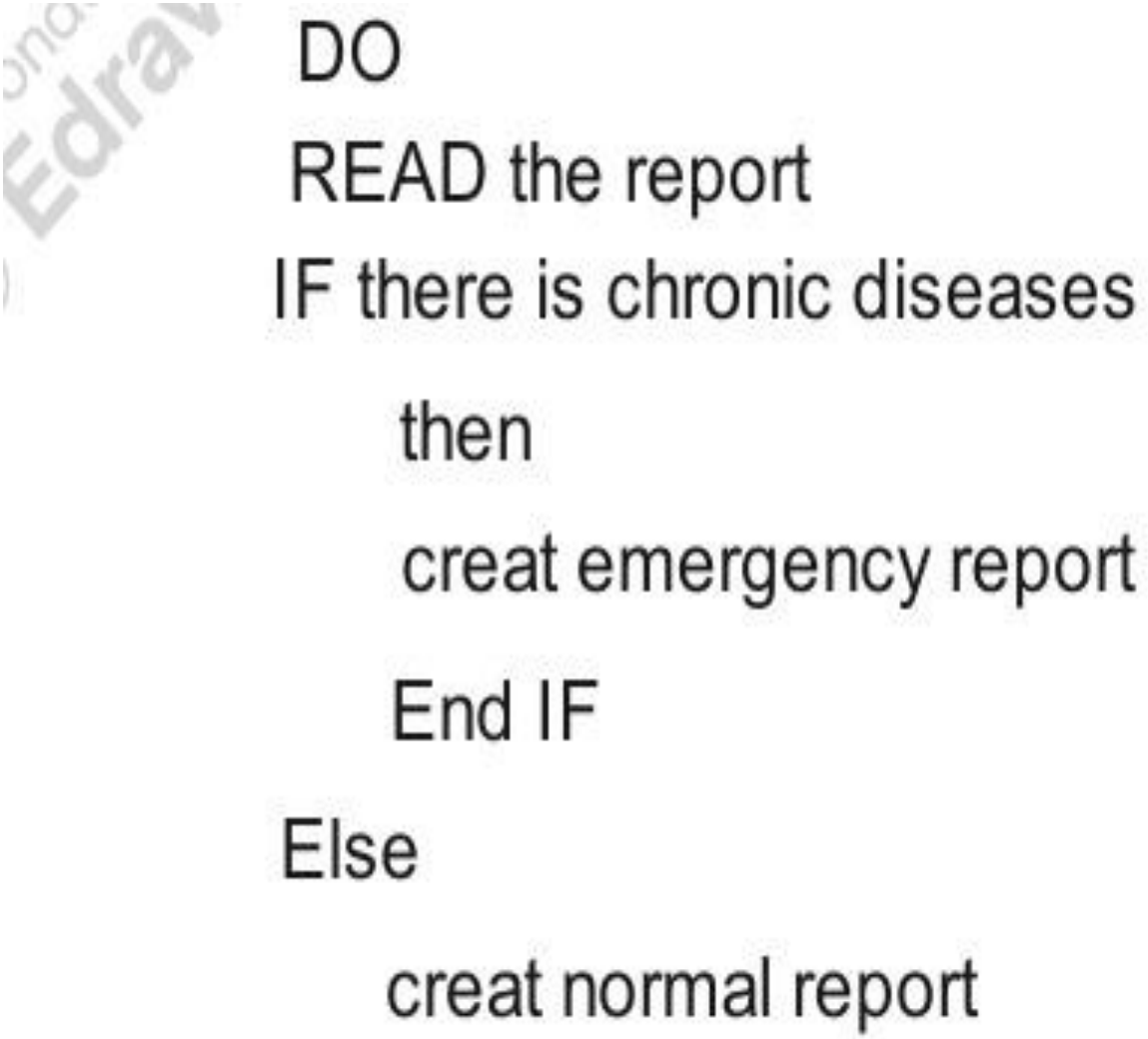
Process 2:- Manage order.



Legend:-

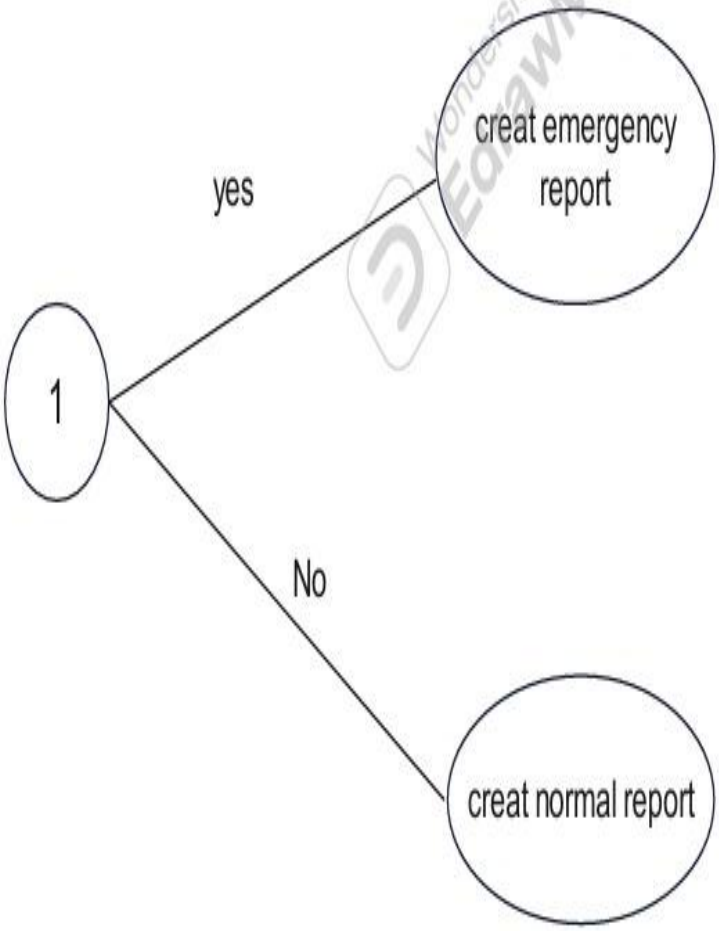
1-Manage Order

Logical modeling for process3:



	3.produce report		
	Condition/Rules	Rules	
		1	2
Condition stubs	chronic diseases	T	F
action stubs	emergency report	X	
	normal report		X

3.produce report



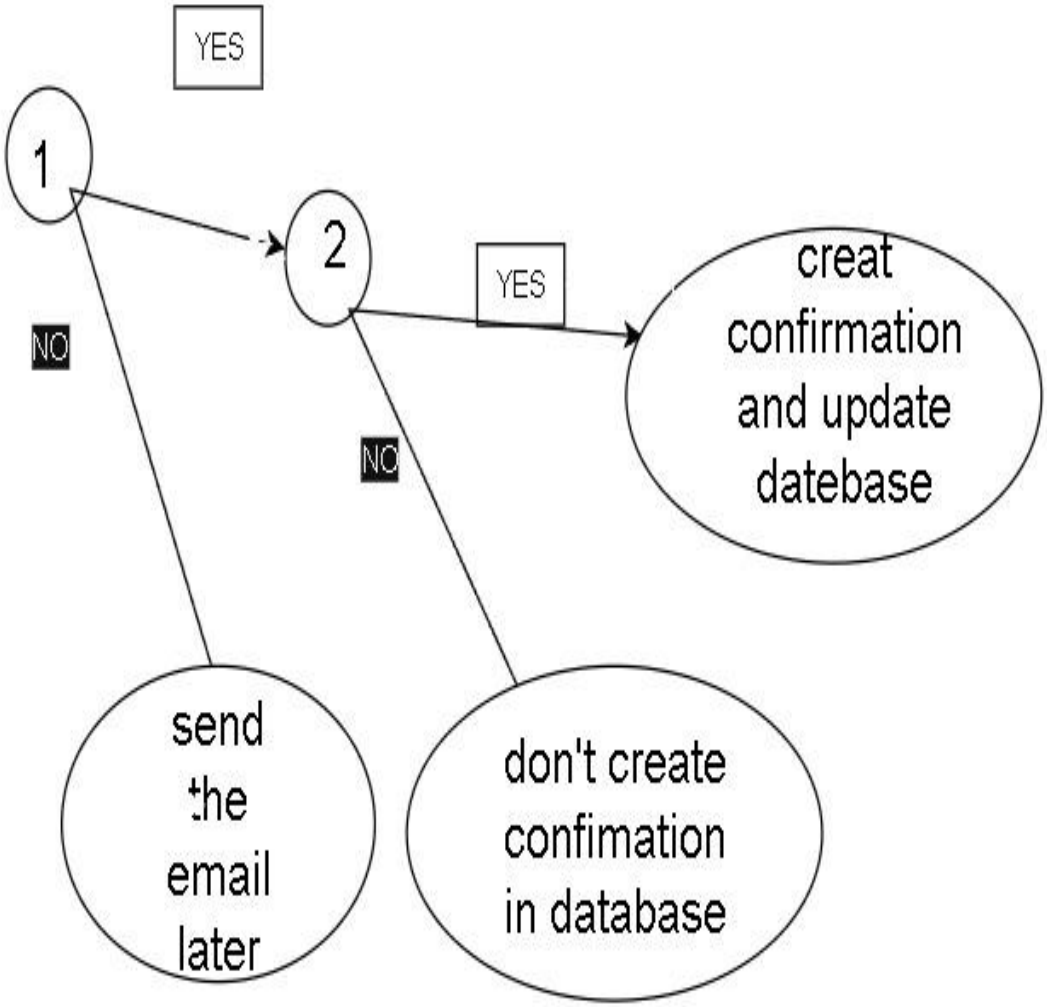
Legend:
1-chronic diseases

Logical modeling for process4:

```
PROCESS Appointment Confirmation Sequence
BEGIN
IF Patient Confirmation Response IS RECEIVED THEN
Create Appointment Confirmation;
Update Database WITH Confirmation Details;
Send Appointment Reminder TO Patient;
ELSE
  Handle case where response is not received

DO NOTHING;
ENDIF;
END
```

Conditions	Rule 1	Rule 2
Confirmation Email Sent	Y	Y
Patient Response Received	Y	N
Create Confirmation	X	
Update Database	X	
Send Reminder to Patient	X	
Do Not Send Reminder (Yet)		X



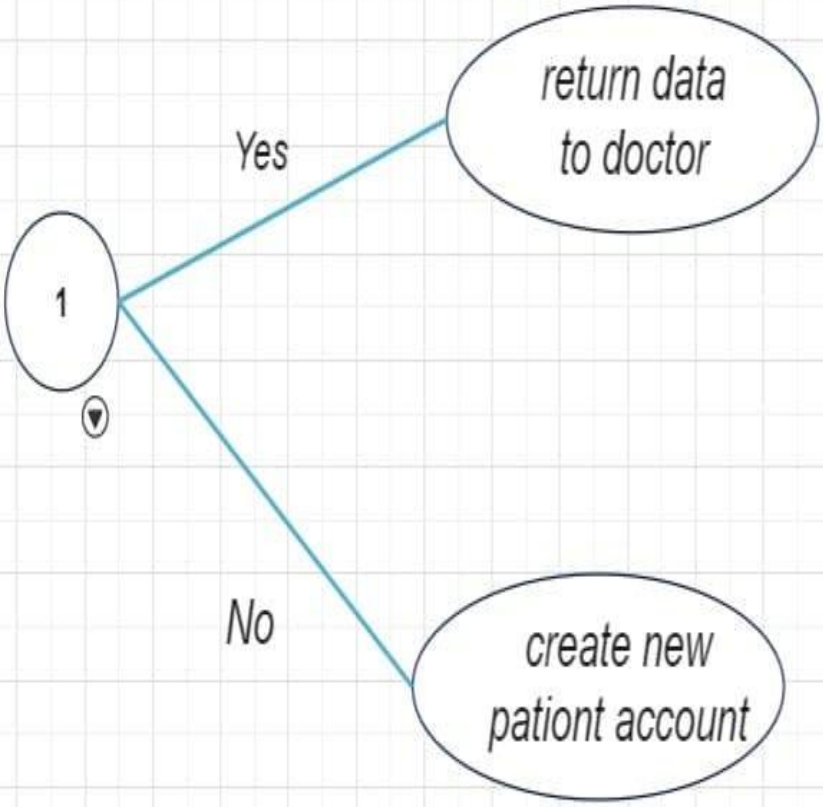
1.confrimation email sent
2.patient response receved

Logical modeling for process5:

```
Do
  Read Request data
  IF there is a patient account
    then
      Return data to doctor
 End IF
Else
    create new patient account
```

	Dission table :- Maintanance data		
	Condition/Rules	Rules	
		1	2
condition stubs	First Visit	T	F
action stubs	Return data to doctor	X	
	Create new pationt account		X

5.0 :- Maintainance data



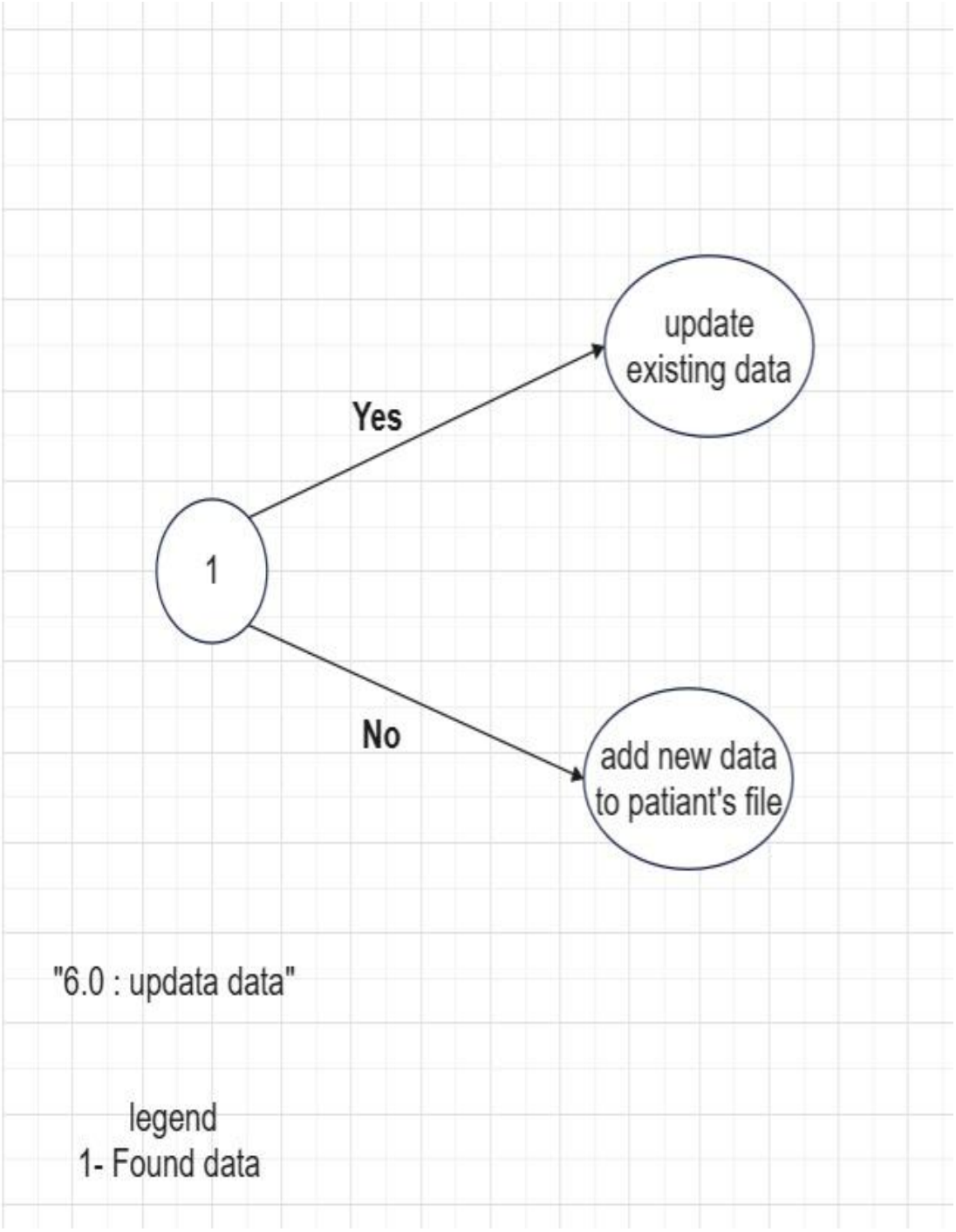
Legend:-
1-request data

(Tree)

Logical modeling for process6:

```
Do
   Check if patient data is found
   IF data is found
      then
         Update existing patient data
   ELSE
      Add new data to patient's file
   END IF
```

• update data			
Condition / Rules		Rules	
		1	2
Condition stubs	Patient data is found	T	F
Action stubs	Update existing patient data		X
	Add new data to patient's fill	X	



"6.0 : updata data"

legend
1- Found data