3.1 Introduction to System Design

A computer is one of the greatest inventions ever made by man. It makes people work easier and faster. The complicated task is able to be done in short time with a good result. Computerization means to change over from a manual system to a computer base system. Computerization of a system involves comprehensive study of existing system, along with its weakness and drawbacks, suggesting a suitable computer system, implementing the new system and testing to its proper functioning.

Another advantage of computer-based systemic is that accuracy in reliability. The process of data provides various type of information. That is why commercial and scientific organization prefers computer based system. As with most real world activities, there are numerous benefits of using a computer system for patient's registration and treatment System for patients.

Another benefit of a software system is the use of a central database. This database is the basis for all actions in the system and can be trivially updated and used to add in all of the system's processes, meaning all of the required information is stored in one central location and thus is easily accessible. This is more reasonable storage method than a paper-based file system, where the time of traveling to and physically searching the records for the required information could be a burden. Human error could also be a factor in that mistakes could be made in the filing process which would not occur in a well written database system and mistakes or changes on physical records can be easy to correct.

3.2 Proposed System and its Features

The new system is proposed after conducting a detailed view of the existing system, and analyzing the problems of the existing system. The proposed computerized system will resolve the problem faced by the users in extracting materials of his choice and helps them to get information. It is an efficient and error free system.

3.2.1 Objectives of the Proposed System

The aim of the proposed system is to reduce the limitations of the current existing system. The requirements for the system have been gathered from the defects recorded in the past.

Following are the objectives of the proposed system:

- **Cost cutting**

  It reduces the cost involved in the patients registration and appointments of doctor.

- **Operational efficiency**

  It improves the operational efficiency by improving the quality of the every process.

3.2.2 Advantages of the System

Advantages of the system are given below as .

- **Processing Speed**

The use of automated method almost invariably speeds up the flow of work in getting information.

- **Efficiency**

The use of automated method improves the efficiency.

- **Flexibility**

The Proposed system allows changes as well as enhancement if working in order to incorporate future needs of the Organization.

- **User Friendly**

The proposed system provides a very efficient, easy and user friendly environment.

- **Performance**

The significant reduction of clerical tasks will lead to much improved staff performance.

- **Usability**

The goal of the proposed system is to keep everything nice, simple and secure for office use.

- **Reliability**

The proposed system is more reliable than the manual one due to its accuracy and security.

- **Accurate**
- In the traditional way human error might creep in due to fatigue and monotonous nature of work involving numbers.
- Very accurate and reliable the process is very accurate and reliable due to limited human involvement because only the authorized user can use the system.

3.3 Unified Modeling Language (UML)

"The Unified Modeling Language (UML) is a language for specifying, visualizing, and constructing the art facts of software systems as well as for business modeling". Visual Modeling is a way of thinking about problems using models organized around read world ideas. Models reduce the amount of time it takes to learn and it improves safety, performance and consistency and to reduce errors. It communicates effectively with process, maps, charts and diagrams of all types.

The UML offers standard semantic and notations for describing object structure and behavior and have emerged as design medium of choice for developing large scale distributed object applications. Rational unified process an extensive set of software development guidelines and rational rose visual modeling tool, the UML, greatly facilitate.

The UML notation is useful for graphically depicting the object oriented analysis or design model. It not only allows specifying system requirements and capturing design decisions, but also promotes communication among key persons involved in the development effort. Rational rose supports the development of the majority of these models as follows:

- Use Case Diagrams

- Sequence Diagrams
- State Transition/Activity Diagram
- Class Diagram
- Deployment Diagram

3.3.1 Use Case Diagrams

The main ingredients for this type of diagram are use cases and actors, respectively the roles that users can take towards a system. It is often used in early stages of design process to collect the intentional requirements of a project. This diagram shows the overall functionality of the system.

Use Case diagrams show the interaction between use cases, which represent system functionality, actors and the people or systems that provide or receive information from the system. Use cases represent the requirements of the system from the user's perspective. So, use cases are the functionality that the system provides. Actors are the administrator and the user of the system.

Use Case Categorization

This describes the importance of the function to the system.

- Primary - These functions are required and are common main processes.
- Secondary - These functions are secondary to the system or rarely occur. Don't need these functions in this iteration. This type of use case is rarely done.

Use Case Description Level (Abstraction)

Essential - A general description of the business process. Do not include technology information. Use the 100 year rule where the information would be understood 100 years in the past and the future.

Real - Design oriented, shows reports, examples. Uses technological descriptions. Real use cases are undesirable during analysis and should only be used during analysis for specific reasons. Real use cases are handy for requirements gathering.

Normally high level essential use cases and expanded essential use cases are done during the analysis phase of a project. A high level real use case is rarely done and an expanded real use case is done during the design phase only if necessary. During the analyses of propose system following Use Cases were identified:

3.3.2 Context Use Case Diagram

Context use case diagram is just same as the simple use case diagram, with a difference that in context use case diagrams are all actors with their functionality within a system are combined instead of separate one.

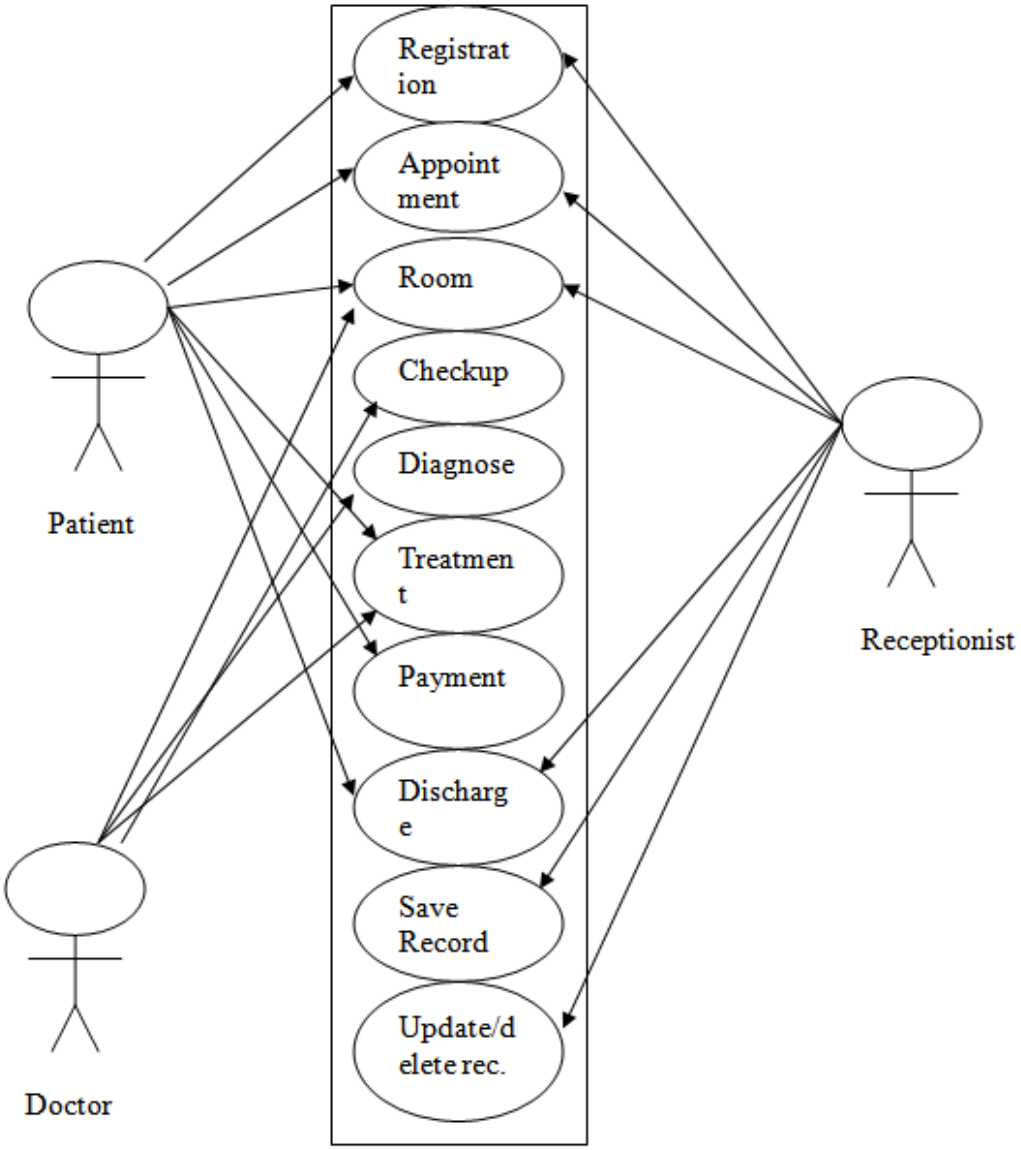


Fig. 3.3.2.1 The context use case diagram of the system is as follows

3.3.3 Sequence Diagrams

The UML system sequence diagram (SSD) illustrates events sequentially input from an external source to the system. The SSD will define the system events and operations. System sequence diagrams are a timeline drawing of an expanded use case.

Events are related by time with the top events occurring first. System events are the important items. These are events that cause a system response.

UML sequence diagrams are used to represent or model the flow of messages, events and actions between the objects or components of a system. Time is represented in the vertical direction showing the sequence of interactions of the header elements, which are displayed horizontally at the top of the diagram.

Sequence diagrams are used primarily to design, document and validate the architecture, interfaces and logic of the system by describing the sequence of actions that need to be performed to complete a task or scenario. UML sequence diagrams are useful design extract from static diagrams or specifications

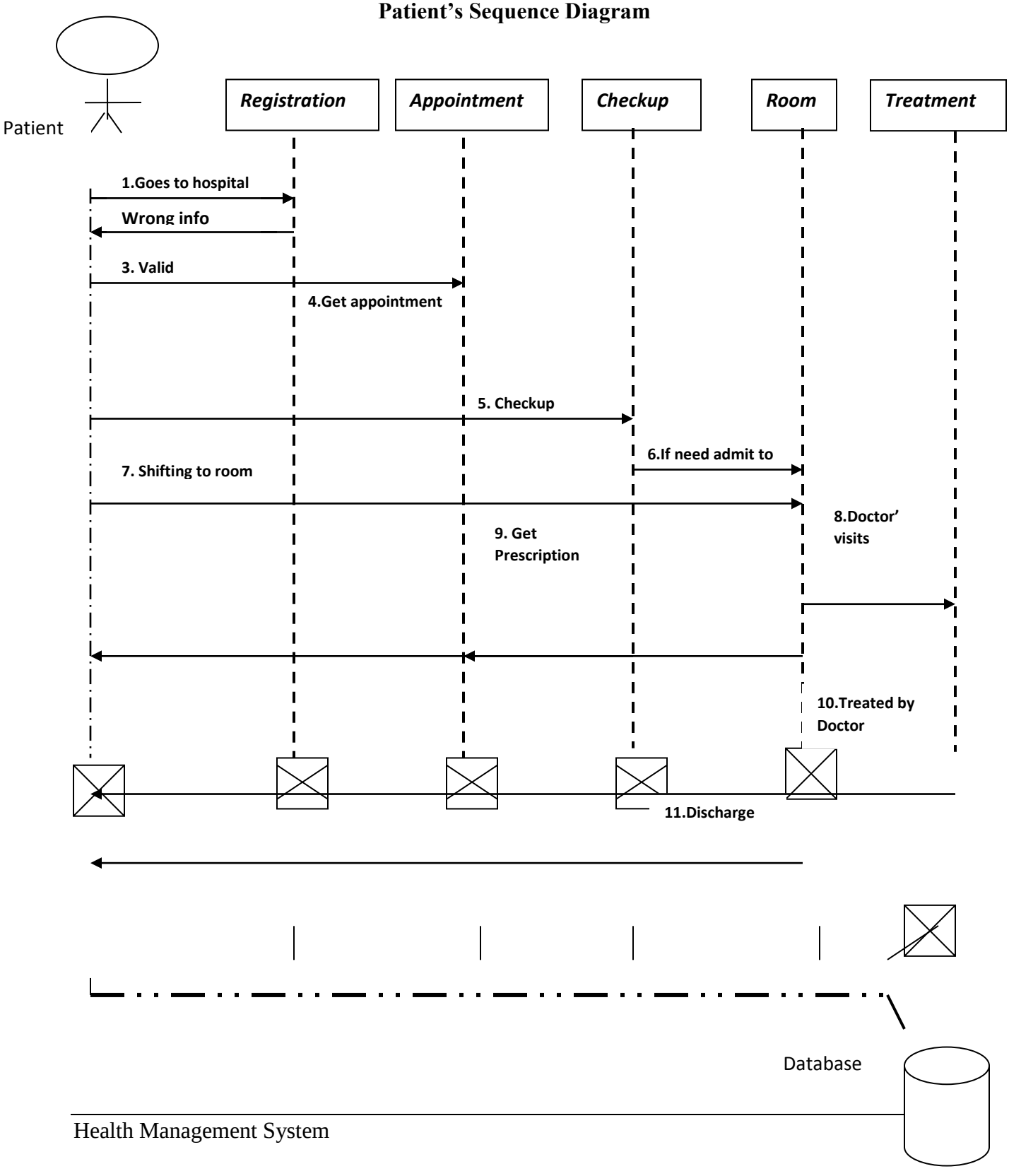
Sequence diagrams are typically used to model:

1. **Usage scenarios** A usage scenario is a description of a potential way your system is used. The logic of a usage scenario may be part of a use case, perhaps an alternate course. It may also be one entire pass through a use case, such as the logic described by the basic course of action or a portion of the basic course of action, plus one or more alternate scenarios. The logic of a usage scenario may also be a pass through the logic contained in several use cases. For example, a student enrolls in the university, and then immediately enrolls in three seminars.
2. **The logic of methods** Sequence diagrams can be used to explore the logic of a complex operation, function, or procedure. One way to think of sequence diagrams, particularly highly detailed diagrams, is as visual object code.
3. **The logic of services** A service is effectively a high-level method, often one that can be invoked by a wide variety of clients. This includes web-services as well as business transactions implemented by a variety of technologies such as CICS/COBOL or CORBA-compliant object request brokers (ORBs).

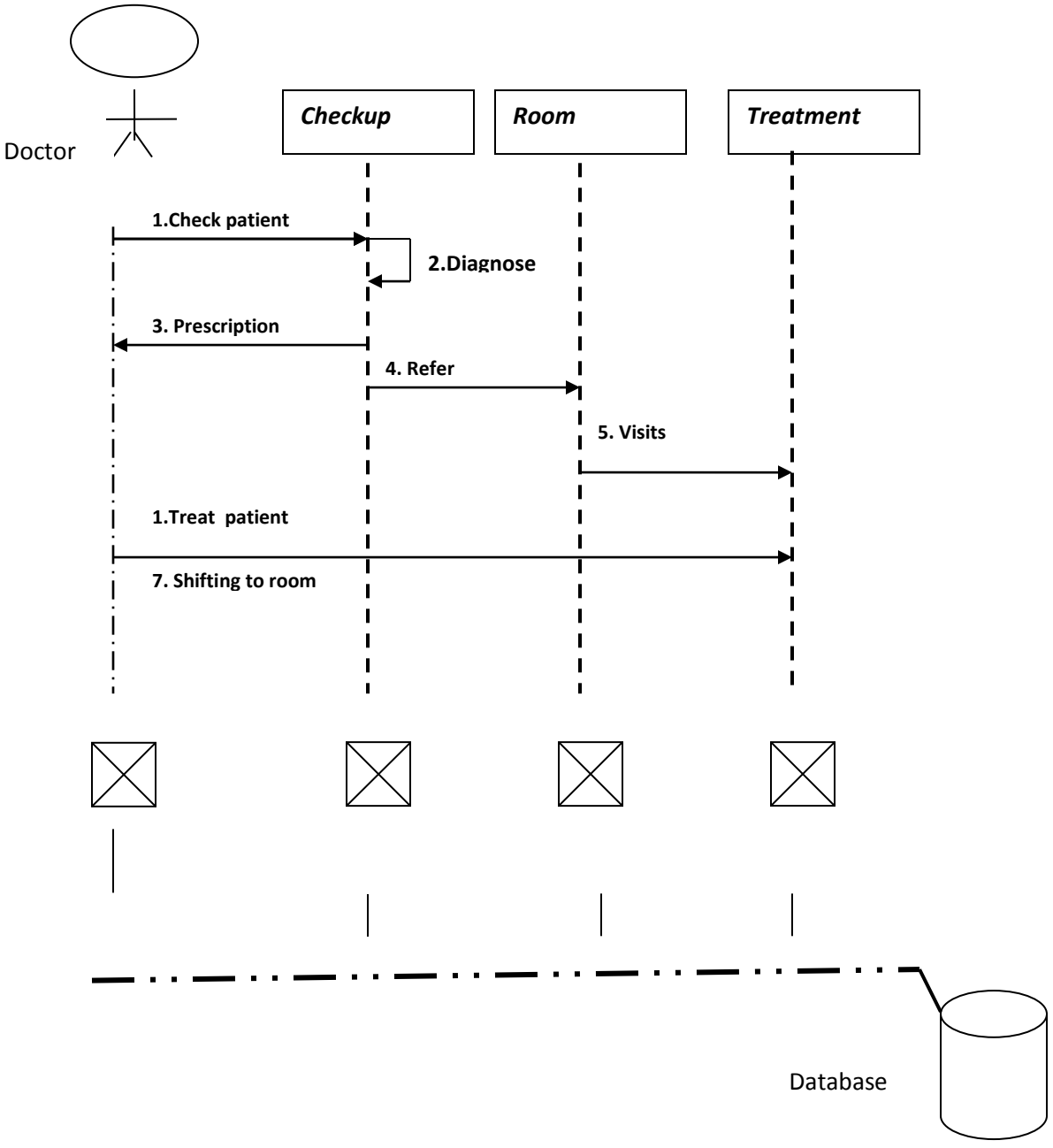
   Use case text may be placed on the left side of the system sequence diagram if desired. If this is done it is best if the use case information lines up with the events in the system sequence diagram.

There may be more than one actor to the system. An actor may be an external automated system that the system may communicate with. Automated actors or robots are shown as actors with a line horizontally through the head.

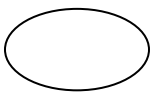
Patient's Sequence Diagram

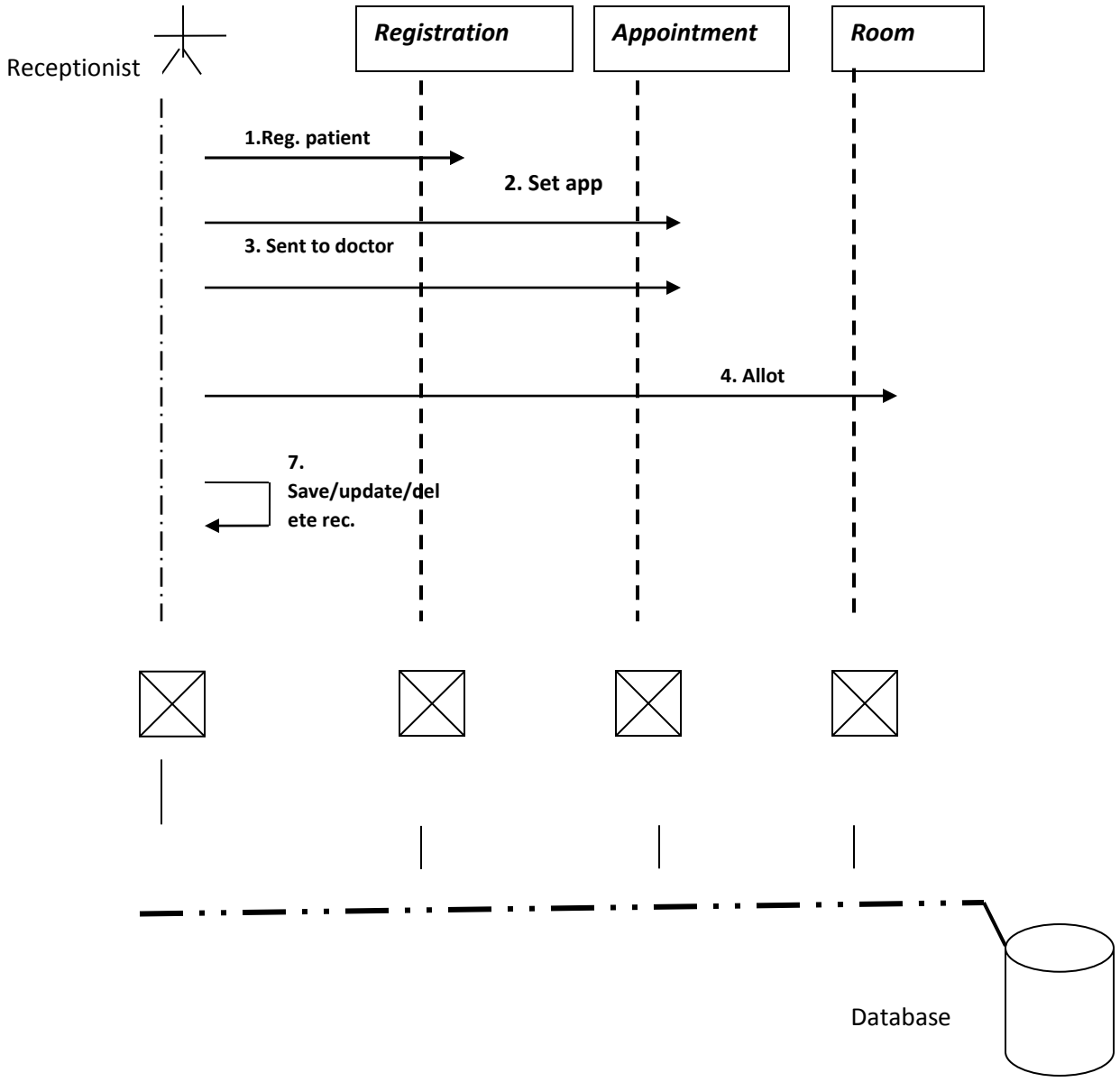


Doctor's Sequence Diagram



Receptionist's Sequence Diagram





3.4 Database Designing

A database is an application that manages data and allows fast storage and retrieval of that data. A database is a collection of data, which is organized in such a way that each piece of data is available to those who need it and with minimum duplication of data. There

are different types of database but the most popular is a relational database that stores data in tables where each row in the table holds the same sort of information. In the early 1970's E.F Cod an IBM researcher derived 12 laws of normalization. These apply how the data is stored and relations between different tables.

Advantages of Database

The most common advantages of database may be as follows.

- Minimal data redundancy
- Consistency of data
- Data integrity
- Sharing of data
- Ease of application development
- Uniform security, privacy and integrity controls
- Reduced program maintenance
- Improved data security

❖ **Entity**

Any object of concept identified by an enterprise that exists independently and about which it is necessary to store data. It may be anything like a person, a place, an event or concept or an object.

❖ **Attributes**

An attribute is the Characteristics or property of an entity that is the interest to the organization. For example, patients is an entity; its attributes mostly may be patient's id, name, and address etc.

❖ **Keys**

A key is a distinct for each individual entity in an entity set. Key attributes are the attributes whose values are uniquely identified and do not exist again.

Different Keys are as follows:

1. Super key

A super key is an attribute or a set of attributes that uniquely identifies an entity. For example, PID is a super key because it can be used to identify each patients uniquely.

2. Primary key

The primary key is the successful client key, the one actually chosen. It may be a single attribute or composite key. The term secondary key is used to mean alternate key but secondary key usually means an attribute or set of attributes whose values not necessarily unique, are used as a mean of accessing records.

3. Foreign key

A foreign key is an attribute or combination of attributes of an entity that is primary key of another entity.

3.4.1 Entity Relationship Diagrams (ERDs)

Entity relationship diagram shows the interactions between classes in the system. Classes contain information and behavior of that acts on that information. Each class in entity diagram is represented by a rectangle divided into three sections:

i. first shows the class name
ii. second shows the attributes
iii. And third shows the actions/ operations of that class.

The lines connecting these classes show the communication/relationship between these classes. The Entity relationship diagram of the running system is as follows;

Entity Relationship Diagram

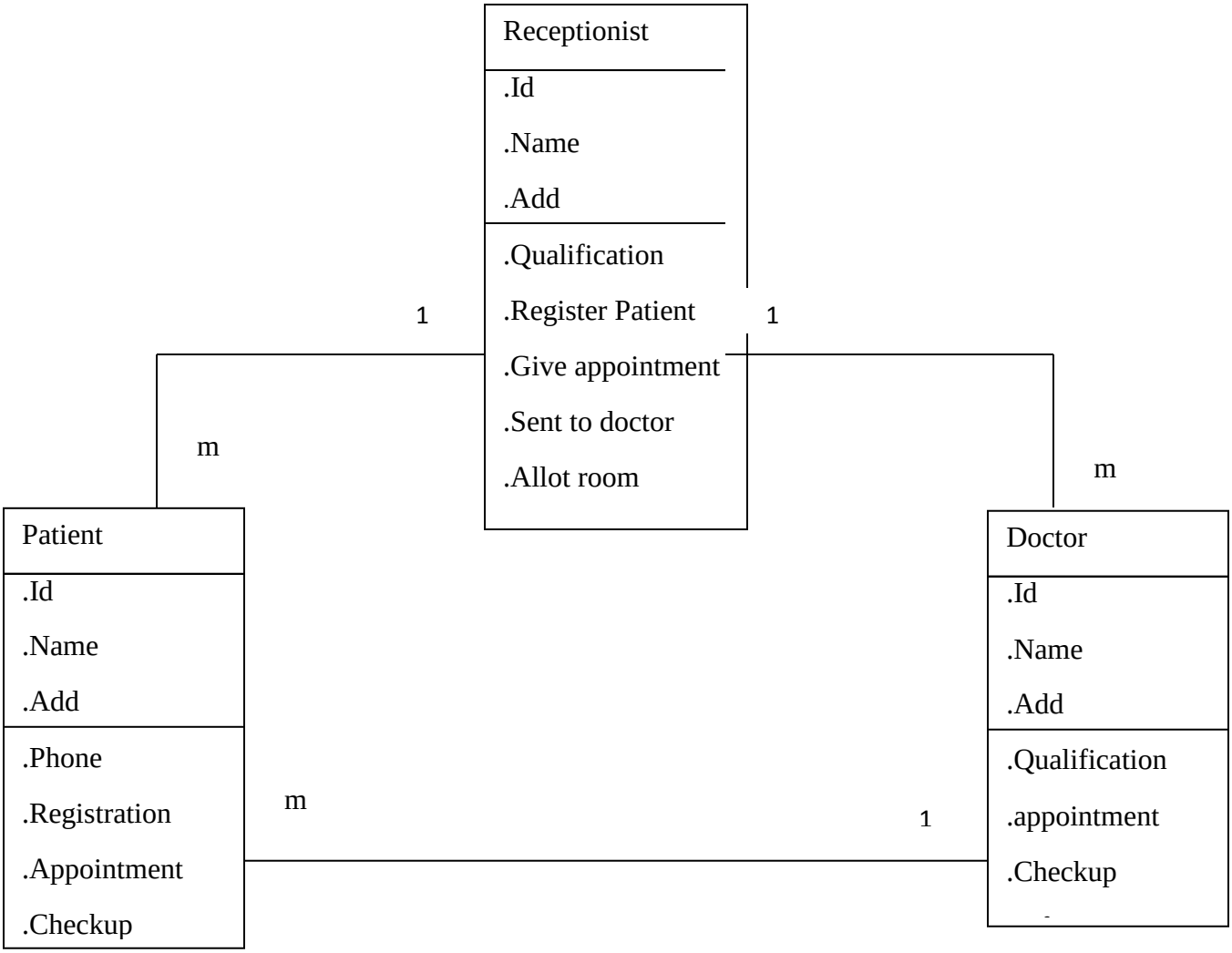


Fig. 3.4.1 entity relationship diagram

3.4.2 Normalization

A relation is in specified normal form if it satisfies the set of requirements or constraints of that form. The major concept used from the relation data model, used in the developing the conceptual model in this system, is the normalization process. Infect normalization process is the process of grouping the data elements. Its simplest definition will be "The process of converting complex data structure into simple stable data structure". Anomalies are the errors or inconsistencies that may result when a user attempts to update a table that contains the redundant data. These types of anomalies are insertion, deletion and the

modification. So another definition of normalizes will be: "Normalization is the process through which we remove insertion, deletion and updating of anomalies of the database."

The definition of the three normal forms is given below:

- **First Normal Form (INF)**

"A relation is in first normal form if every cell of a table must contain an atomic value".

For example in this project patients & appointment date have one or many previous education records that's why a new table is made to overcome the problem and the data is in 1st Normal form.

- **Second Normal Form (2NF)**

A relation is in second normal form if it is in first normal form and every non-key attributes fully functionally dependent on the primary key. For example, in this project table of person info has PID and PName as a composite key and all non-key attributes are fully functionally dependent on this key that's why the data is in 2nd Normal form.

- **Third Normal Form (3NF)**

A relation is in third normal form if and only if it is in second normal form and no non-key attribute is transitively dependent on the primary key. It is also stated as "A relation is in third normal form, if it is in second normal form and no transitive dependencies exist". In this project all tables are in 1st and 2nd normal forms and no non-key attribute is transitively dependent on the primary key.

- **Higher Normal Forms**

There are several higher normal forms, including 4NF, 5NF, BCNF, and PJ/NF. We'll leave our discussion at 3NF, which is adequate for most practical needs.

3.4.3 Database Tables

The following tables are maintained to fulfill the requirements of the proposed system. The table name and the field name with its type are also given below with the description of each. The primary key is identified by the symbol of key with that attribute.

Patient File

Data file name	patient
Primary key	patient_id

Record Layout

Field Name	Description	Type	Length
patient_id	Patient Id	int	10
p_name	Patient Name	varchar	50
p_add	Patient address	varchar	50
p_cnic#	Patient CNIC#	varchar	50
p_phone	Patient Phone#	int	15
p_age	Patient Age	int	10

Explanation

This file holds all the information of the patients. patient_id is the primary key of the patient file. It is also indexed on p_name. With the help of p.k, the record for any required patient can be accessed quite easily. This file is linked with the files Appointment File, Room-patient File.

The link field is patient_id.

Doctor File

Data file name	doctor
Primary key	doctor_id

Record Layout

Field Name	Description	Type	Length
doctor_id	Doctor Id	int	10
doc_name	Doctor Name	varchar	50
doc_add	Doctor address	varchar	50
doc_cnic#	Doctor CNIC#	varchar	50
doc_phone	Doctor Phone#	int	15
Specialization	Specialization	varchar	50
Qualification	Qualification	varchar	50
Date-of-joining	Date-of-joining	varchar	50

Explanation

This file holds all the information of the doctor. doctor_id is the primary key of the doctor file. It is also indexed on doc_name. With the help of p.k, the record for any required doctor can be accessed quite easily. This file is linked with the files Appointment File.The link field is doctor_id.

Room File

Data file name	room
Primary key	room_id

Record Layout

Field Name	Description	Type	Length
room_id	Room Id	int	10
Category	Category	varchar	25
Per-day-charges	charges of Rooms/day	int	25

Explanation

This file holds all the information of the rooms. room_id is the primary key of the room file. With the help of p.k, the record for any required rooms can be accessed quite easily. This file is linked with the file Patient File, Room-patient File.

The link field is room_id.

Room-patient File

Data file name	room-patient
Primary key	room_id, patient_id

Record Layout

Field Name	Description	Type	Length
room_id	Room Id	int	10
patient_id	Patient Id	int	10
Status	Status	bit	01

Explanation

This file holds all the information of the rooms and the patient in it. room_id and patient-id are the primary key of the room-patient file. With the help of p.k, the record for any required rooms and the patient in it can be accessed quite easily.

Nurse File

Data file name	nurse
Primary key	nurse _id

Record Layout

Field Name	Description	Type	Length
nurse _id	Nurse -Id	int	10
nurse _name	Nurse Name	varchar	50
nurse _add	Nurse address	varchar	50
nurse _cnic#	Nurse CNIC#	varchar	50
nurse _phone	Nurse Phone#	int	15
Qualification	Qualification	varchar	50
Date-of-joining	Date-of-joining	varchar	50

Explanation

This file holds all the information about the nurses. nurse_id is the primary key of the nurses. File. With the help of p.k, the record for any employee can be accessed quite easily.

Appointment file

Data file name	appointment
Primary key	appointment_id

Record Layout

Field Name	Description	Type	Length
appointment_id	Appointment Id	int	10
doctor_id	Doctor_id	int	10

app_time	Appointment time	time	10
app_date	Appointment_date	date	10

Explanation

This file holds all the information of the appointments. appointment_id is the primary key of the room file. With the help of p.k, the record for any appointment can be accessed quite easily. This file is linked with the file Patient File, Doctor File.The link field is appointment_id.