TABLE OF CONTENTS

ABSTRACT

Application that could use by the customer to manage Best Practice with respect to Client’s Project , Project’s Site and Site’s Workshop. Customizable repository is to be created for the maintenance of the document via internet. It’s a multi-user system and provides an interface between the user and the application. The System will be responsible for the fallowing activities

Admin Module

- Planning Module
- Statistics Module
- Document
- Helps

INTRODUCTION OF PROJECT

There are number of ways in which computer affect human lives which are uncountable. Computer process the function with much faster speed and greater accuracy. So among the various information system of data processing my project “PATIENT BILLING SYSTEM ” one of the data processing system, which mainly concerns with the customer billing record of the hospital made the entire performed are computerized.

The main objective of the project is to hail a fully computerized system for main fining patient detail, patient bills, and ward details.
The project has the following features:

- Effectiveness
- Easy to use
- Consistency
- Simplicity
- Accuracy

The purpose of the Patient Billing System (PBS) is to capture necessary data elements from the agencies' internal bills/personnel systems, validate the information before bills are processed, and post the validated bills and personnel information. This new system is being developed to support the client server efforts. . The main aim of the project is to create automated software which is purely used for billing operation in a hospital. This application would be facilitating the particular authorities of a hospital to generate and update the bills of a patient and store them. This will later be retrieved by the administrator at the time of discharge.

Developing a local patient billing software system would benefit the hospital management. This system basically works for the hospital management, helping them to generate bills and preserve patient’s details in a well-organized approach. Since it is Software driven the quality of services can be enhanced considerably.

A BRIEF OVERVIEW OF THE ORGANIZATION

The purpose of this section is to obtain agreement regarding objectives the system must meet. Ultimately, this segment defines the boundaries of the effort.

Customer: The target people are the hospital management people. The authorities are provided with a user id and they maintain their own accounts. The authorities can even change their passwords.

Purpose: The purpose of this application is to provide complete convenience to the management of the hospital, in order to make Billing, an effortless task to perform. This project helps the administrator to overcome the difficulty in tracking and maintaining records of the patients. It saves his time because the tasks are sub-divided and assigned to respective authorities. So, every authority has only some degree of job to accomplish and hence avoids overload.

Scope: Release 1 focuses on :
Particular Authorities in a hospital.

SYSTEM REQUIREMENTS:

Problem formulation:-

The first step in the system development life cycle is the identification of a need. This is a request to change. Improve or enhance an existing system. The main purpose of this phase problem becomes easy to understand.

Initial Investigation:-

First we went in the hospitals where we done the initial investigation about the hospital. As we have described earlier in the phase of problem relation that there are several types of problem in present working of manual working therefore in the phase of initial investigation we try to find those problems and there solution.

- ***TOO LATE:*** In the hospital all the working was manual. Due to the manual working there was too late.
- Being the all process in manual all the process was very low.

Future scope:-

In future this may work for the manufacture by making some modification in the project models by developing some extra networking feature this “PATINET BILLING SYSTEM” will work in interact very efficiency.

This system would be much more helpful for the near future because it is developed according to the all condition which happened in the future. Like as add new features .it can modify of all record, it is also helpful for updating account of patient record and detail.

ADVANTAGE OF THE SYSTEM:

- The main advantage of the system is the process is very simple like billing of patient.
- We can easily search any record of available patient.
- We have low chances of errors in this system thus it is more accurate. Various constructs can be applied on the software.
- It will be more reliable with incensed throughput capacity and Economy.
- It will save our lots of time because of feat processing as compared to human
    - We can easily maintain patient record.
    - We can easily search the information about particular record with help of various options.

We can easily search the patient information by giving their name or patient id.

HARDWARE/SOFTWARE REQUIREMENT:-

Server configuration:
Minimum 100MB Hard Disk
P-III processor or equivalent
RAM 128 MB
Windows or Linux
Operating System – Linux, Microsoft
Language -- visual basic6.0,
Database --Ms Access

ACCEPTANCE CRITERIA:-

The main aim of the project is to develop a feature-rich, practical Patient Billing Software (PBS) for a hospital. The authorities of a hospital can login into their respective accounts and then update the bills consequently. The main aim of the project is to develop a feature-rich, practical Patient Billing Software (PBS) for a hospital. The authorities of a hospital can login into their respective accounts and then update the bills consequently.

- PBS will Provide a repository for detailed patients' bills and personnel data
- Have a standard input format that is not patient specific
- Validate, through edits, the information reported is properly authorized
- Generate proper & valid transactions to initiate the payment process

PBS is needed to get rid of difficult and tiring conventional model. As we can take a look into conventional model.

CONVENTIONAL MODEL:-

The traditional way of maintaining details of a patient in a hospital was to record them Using case sheets manually. It was difficult to manually retrieve data whenever needed and almost next to impossible to maintain such capacious data as the years passed by. And the billing had become a hard-hitting task for the data operators.

Hence to prevent losing of data and enhance the maintainability, various computerized approaches have been evolved.

NEED FOR AUTOMATION:-

The operations in the manual process are very time consuming and is an overhead process. And through the manual process of maintaining records, it became a chaotic operation for the authorities whenever they wanted to refer any patient's details.

At that point of time, it was really essential to have an automated system which would make work easier and proficient.

PROPOSED SYSTEM:-

With time and growth of programming capabilities, more and more sophisticated, automated patient billing systems have evolved. The proposed Patient Billing System has rectified almost all disadvantages of the existing system. By making this system as automated, the hospital management's are benefited the most.

An Automated computerized Billing Software System provides greater enhanced techniques for maintaining data in a hospital and also helps in easy retrieval and updating of data, which results in efficient billing.

The PATIENT BILLING SYSTEM method aims at improving the human behaviours at work (to improve safety and reduce manual work).

The process is the following:

To initiate the project, a visual basic consultant helps Workshop workers to identify the best practises to follow. This is done for all the workshops. Then, every Workshop involved in the project has a list of best practises to respect.

The BV consultant setups the application by entering the configuration (sites definition, workshops definition, shifts definition for all the workshops, list of best practises by workshop).

At a defined frequency (daily, weekly…), for each shift of the workshop, a user uses the application to generate the list of best practises to follow for his workshop.

The user assesses the best practises respect in the workshop and enters the result into the application.
The application is able to generate graphs showing the result of the assessments (rate of respect for all the best practises and trend of this result during a period)
The graphs generated by this application is used during periodic meetings debrief of the result with the workshop workers and to define which best practises can be added and which practises can be removed.

ADMIN MODULE:-

Through this module user could create update modify and delete in different sub module as follows:
Patient detail
Ward detail
Bill detail

PLANNING MODULE:-

Here user Planning for best practice for a workshop to improving working standard.

STATISTICS MODULE:-

Statistics showing the status which displays the status of bills the within a time period.

ADVANTAGES OF THE PROPOSED SYSTEM:-

The advantages of proposed system can be summarized as below:

(1) The new system will developed in MS Access concept making it compatible with the user requirement. As a result there will be minimum redundancy of data and large amount of data can be stored in the database without any storage problem.

(2) The response time information retrieval will reduced to negligible and the processing time require for retrieving, the desired information from different table is less.

(3) Reports can be generated quickly for effective decision making and up-to-date information are available and answers to queries are also provided instantly.

FEASIBILITY STUDY:-

A feasibility study is defined as an evaluation or analysis of the potential impact of a proposed project or program. A feasibility study is conducted to assist decision-makers in determining whether or not to implement a particular project or program. The feasibility study is based on extensive research on both the current practices and the proposed project and its impact on the school foodservice operation. The feasibility study will contain extensive data related to financial and operational impact and will include advantages and disadvantages of both the current situation and the proposed plan.

The feasibility study is conducted to assist the decision-makers in making the decision that will be in the best interest of the school foodservice operation. The extensive research, conducted in a non-biased manner, will provide data upon which to base a decision. The Feasibility study could be used because:

Main characteristics of the system:

- Manage the information of client, project, site, workshop and shift.
- Setting and improve working standard.
- Provide Graphical representation of data..
- User Access privilege (Profile).

Within a feasibility study, six areas must be reviewed, including those of Economics, Technical, Schedule, Organizational, Cultural, and Legal.

Economic Feasibility Study:-

This involves questions such as whether the firm can afford to build the system, whether its benefits should substantially exceed its costs, and whether the project has higher priority and profits than other projects that might use the same resources. This also includes whether the project is in the condition to fulfill all the eligibility criteria and the responsibility of both sides in case there are two parties involved in performing any project.

Technical Feasibility Study:-

This involves questions such as whether the technology needed for the system exists, how difficult it will be to build, and whether the firm has enough experience using that technology. The assessment is based on an outline design of system requirements in terms of Input, Output, Fields, Programs, and Procedures. This can be qualified in terms of volumes of data, trends, frequency of updating, etc. In order to give an introduction to the technical system.

Schedule Feasibility Study:-

This involves questions such as how much time is available to build the new system, when it can be built (i.e. during holidays), whether it interferes with normal business operation, etc.

Organizational Feasibility Study:-

This involves questions such as whether the system has enough support to be implemented successfully, whether it brings an excessive amount of change, and whether the organization is changing too rapidly to absorb it.

Cultural Feasibility Study:-

In this stage, the project's alternatives are evaluated for their impact on the local and general culture. For example, environmental factors need to be considered.

Legal Feasibility Study:-

Not necessarily last, but all projects must face legal scrutiny. When an organization either has legal council on staff or on retainer, such reviews are typically standard. However, any project may face legal issues after completion too

Marketing Feasibility Study:-

This will include analysis of single and multi-dimensional market forces that could affect the commercial.
So, before developing this system our team will generate some document related to feasibility study, which include all such feasibility analysis as describe above.

EFFORT ESTIMATION:-

The project estimate is only as good as the estimate of the size of work to be accomplished; choosing a right software sizing metric is an important task.

In the context of project planning, size refers to a comfortable outcome of the software project. If a direct approach is taken, size can be measured in LOC. If an indirect approach is chosen functional measures are considered.

The LOC is an artifact of all software development projects and can be easily counted. Even though, it is not universally accepted as the best way to measure the process of software development.

Loc measures are programming language independent.
They penalize well designed but shorter programs
They cannot easily accommodate non procedural languages.
Their use in estimation requires a level of detail that may be difficult to achieve.(i.e., the planner must estimate the LOC to be produced long before analysis and design have been completed)

The functional Point Metrics on the other hand measures the functionality delivered by software and a measure about the functionality that an application delivers to the users.
Functional points stay constant regardless of the programming languages used.
Functional points can be developed relatively easily by consulting the users at an early stage of the development process.
The steps taken to arrive at the functional point count for a software product are:

The information processing size , is determined by identifying the system components perceived by the end users and then classify and count the five user function types (known as unadjusted function points) delivered by the development process.

Unadjusted Function Points
These are classified as follows:

Data Function

Number of Internal Logical Files (ILF): Logical grouping of data in a system, maintained by an end user, are referred to as internal logical files.

Number of External Interface Files (EIF): All machine readable interfaces that are used to transmit information to another system.

Transactional Function: Number of External Inputs (EI): An external Input gives the user capability to maintain the data in ILFs through adding, changing and deleting its contents.

Number of External Outputs (EO): Each user output that provides application oriented information to the user. In this context output refers to reports, error messages, and so on. Individual data items within a report are not counted separately.

Number of External Inquiries (EQ): An inquiry is defined as an online input that results in the generation of some immediate software response in the form of an online output.

These components are further categorized as simple, average or complex, depending on the number of data elements in each type and other factors.

Degree of influence (DI) of fourteen components called ‘general application characteristics’ is determined. The degree of influence of each of these factors takes a value from 0-5 to signify none to essential. The sum of the scores of the fourteen characteristics, that is the total degrees of influence, is converted to the ‘technical complexity factor’ by using the following formula:

TCF = 0.65 + 0.01 x DI

The relative system size expressed in function points (FPs) is computed as:

FP’s = UPF x TCF

For the Proposed Project the details are as stated-below:
(Note: The below stated numbers have been concluded with reference to the Requirements document, described earlier)

Number of External Inputs:	22
Number of External Outputs:	40
Number of External Queries:	18
Internal Logical Files:	4 Files
External Interface Files:	1 File

COMPUTATIONS:

Measurements Parameters	count	weighing factor simple average complex	
EI	22	x3 **x4** x6	=88
EO	40	**x4** x5 x7	=160
EQ	18	x3 **x4** x6	=72
ILF	4	**x7** x10 x15	=28
EIF	1	**x5** x7 x10	=5
COUNT UFP TOTAL			=353

COMPUTING TECHNICAL COMPLEXITY FACTOR(TCF)

CI	CHARACTERISTIC	DI	COMMENT
C1	Data communication	2	Client/server architecture
C2	Distributed functions	3	Client/server Architecture
C3	Performance	2	Client/server architecture
C4	Heavily used configuration	1	Application design
C5	Transaction rate	5	Strong influence for transaction handling
C6	Online data entry	4	Majority operations depends on online entries
C7	End user efficiency	**3**	Exclusivity is sprawled over user friendly menu
C8	Online update	**2**	Moderate influence
C9	Complex processing	**4**	Database functions.
C10	Re-usability	**3**	Modules must be re-usable for further enhancements
C11	Installation Ease	**3**	
C12	Operational Ease	**5**	
C13	Multiple Ease	**5**	
C14	Facilitate Change	**3**	
TOTAL DI		**45**	

Degree of Influence	**Values**	**Degree of Influence**	**Values**
Not Present	0	Average Influence	3
Insignificant Influence	1	Significant Influence	4
Moderate Influence	2	Strong Influence, Throughout	5

Technical complexity as defined above:

TCF = 0.65 + 0.01 x DI

Computed as:
TCF = 0.65 + 0.01 x 45

TCF = 1.1

Function point defined as:

FP = UPF x TCF

i.e. FP = 353 x 1.1
FP = 388

For Object Oriented Technologies LOC/FP = 30

LOC = FP x 30
LOC = 388 x 30
LOC = 11649 DLOC

- Expected lines of code to be developed: 12 KDLOC Approx.
- Expected scheduled delivery time: 1.8 * man-months period.

(From company's empirical data, the similar projects based on Visual Basic and MS Access it is assumed that the project must complete within above specified period of time.)

TIME ESTIMATION-

According to COCOMO Model-

Estimation of Development Effort-

Organic : Effort=2.4(KLOC)$^{1.05}$ PM

In this project KLOC as estimated above=11649
Therefore, Effort = 2.4*(11649)$^{1.05}$
Effort = 15.181

Estimation of Development Time--

Organic : T_{dev}=2.5(Effort)$^{0.38}$ Months

As estimated above, Effort = 15.181
Therefore, T_{dev} = 2.5*(15.181)$^{0.38}$
Estimated Time = 5 Months Approx.

(From company's empirical data, the similar projects based on Visual Basic and MS Access it is assumed that the project must complete within above specified period of time.)

COST BENEFITS ANALYSIS:-

The costs associated with the system are expenses outlay or losses arising from developing and using a system. But the benefits are the advantage received from installing and using. Cost and benefits can tangible or intangible, fixed or variable, direct or indirect.

Tangible or Intangible cost:-

Tangible cost is that which value can be measured. The outlay of cash for any specific item or activity is referred to as a tangible cost. These costs are known and be estimated accurately. Cost that are known to exist but their financial value can not be exactly measured are referred to as intangible cost.

Cost of the system	
One PC core 2 due	36000/Rs
One inkjet printer	4500/Rs
Total Hardware costs	**40500/Rs**

Cost of the SOFTWARE	
Visual basic 6.0	10000/RS
Ms Access	5000/RS
Total software cost	**15000/RS**

Development cost:	
Programming	20000/Rs
For Analyst	5000/Rs
Total development cost	**25000/Rs**

Intermediate printouts	150/Rs
Electricity cost	500/RS
Others	100/Rs
Total cost	**650/Rs**

COST ESTIMATION OF PROJECT:-

This is the full-fledged industrial project containing database handling, active documents, good designing, validation checks, good security and easy to use interface etc.
In this project I have used latest s/w technologies. Visual basics 6.0 is the front end and used, ms access is used as database (back end), operating system windows vista ultimate.
Total Effort Estimation = 15.181
Nominal Development Time =5 Months
Salary of Programmer =12,000
Cost required developed the product=60,000 Approx

DIRECT AND INDIRECT COSTS:-

Direct costs are those which are directly associated with a system. Its are applied directly to operator of the site.

Indirect costs are not directly associated with a specific activity in a system. It is often refers to as overhead expenses.

DIRECT OR INDIRECT BENEFITS:-

Direct benefits also can be specially attributes to a given project.
Saving of time is writing the record and transforming the old data from the store.
Saving man power.
Indirect benefits are realized as a byproduct of another system for example:

- Easy entry of record
- Easy to modify any record
- Easy to calculate feedback and determine weak points of any system.
- Easy access of information with specific requirements.
- Information can access from any other site of same objective.

Request Period:-

It is not necessary that all the required projects are desirable or feasible, because requested projects only initialize when they are feasible in each and every aspect. If the cost is more than the cost of benefits after the project release then the project request are dropped. Under this feasibility study the user system "Patient Billing System" are feasible in each and every aspects. In other words we can say that the end user system is technically feasible economically feasible operational feasible and legally and time feasible.

SCHEDULE:

Development Schedule

MODULES	START DATE	END DATE
A. Documentation	15-sep-2011	4-oct-2011
1. SRS	7-oct-2011	15-oct-2011
2. HLD	29-oct-2011	30-oct-2011
3. LLD	20-oct-2011	21-oct-2011
4.Test Plan	20-nov-2011	28-Nov-2011
5.Test Case Report for all modules	28-Nov-2011	29-nov-2011

PROGRAMMING LANGUAGE AND DEVELOPMENT TOOL:-

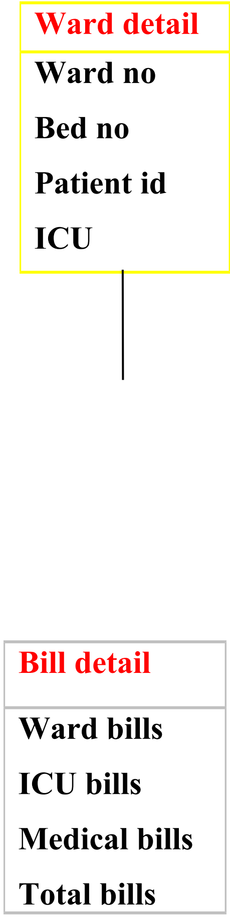
Programming language: ***Visual basic 6.0,Ms Access. Operating system: windows vista,etc.***

Patient Billing System:-

Product Perspective:-

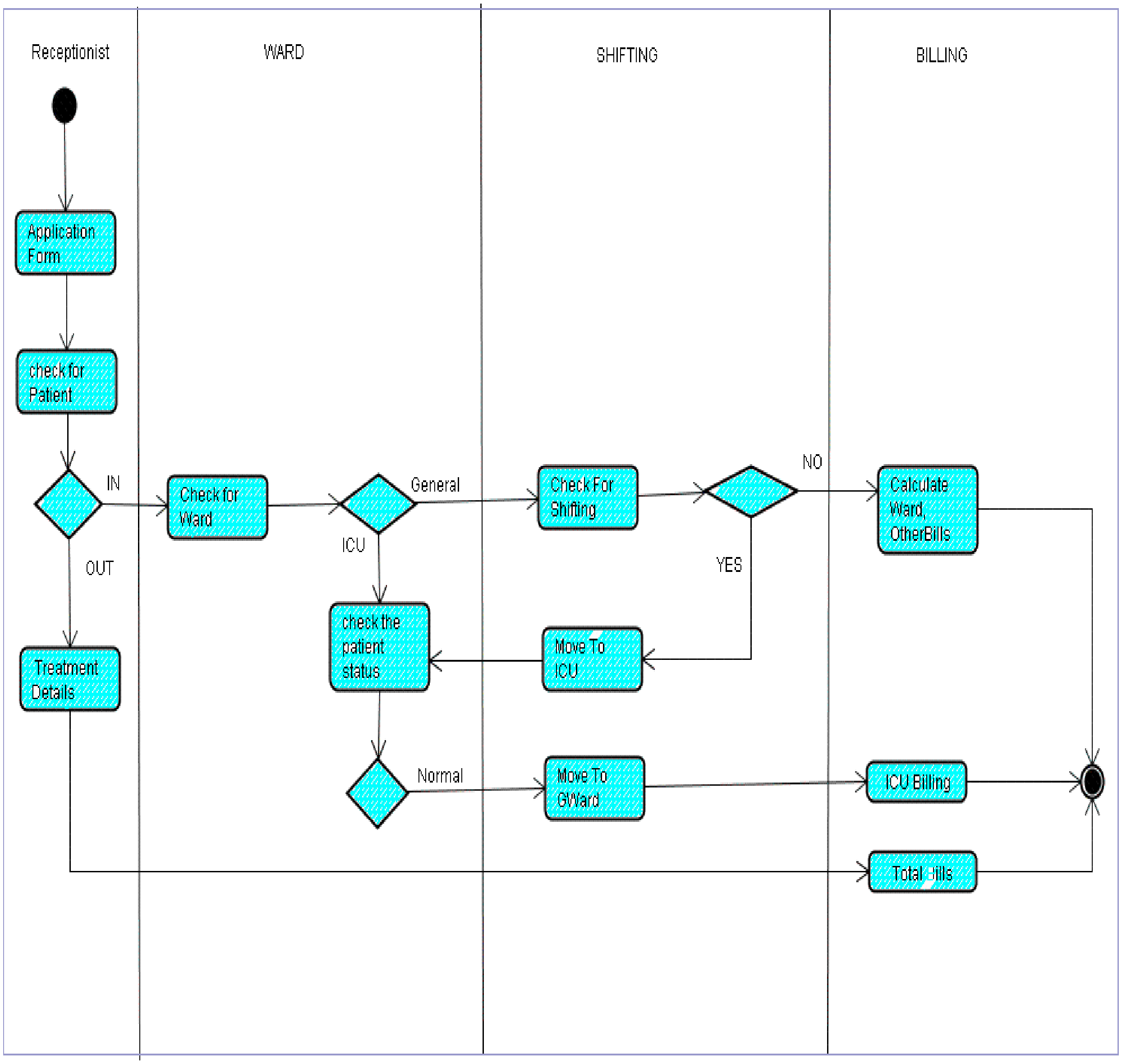
PBS collects and stores bills and personnel data in a database on the Comptroller's mainframe. The system applies edits to validate data before accepting a transaction. The following diagram depicts the processes contained within PBS:-

Class Diagram :

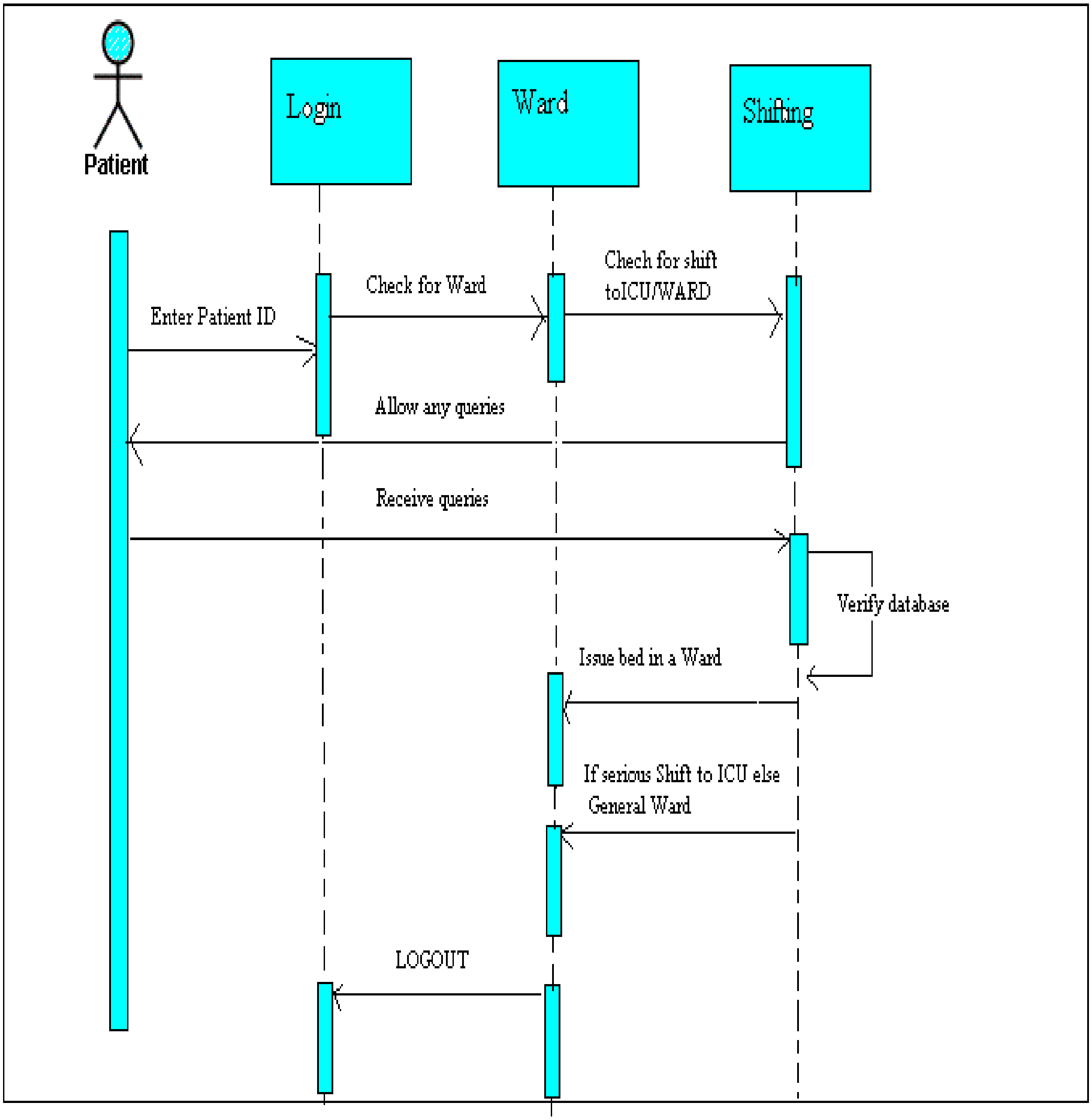


PRODUCT FUNCTIONS:-

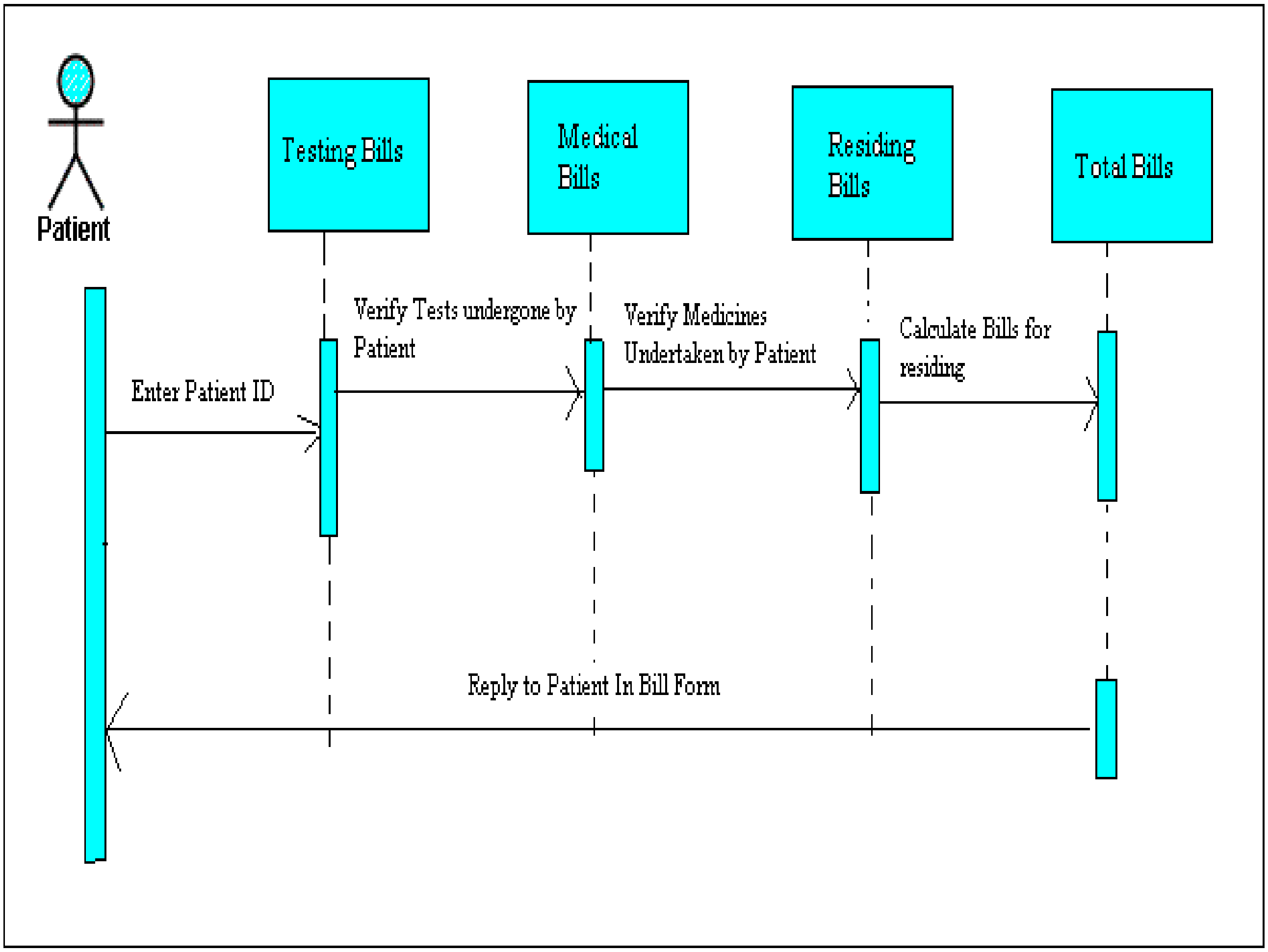
The PBS functional architecture is shown in the figure below. There are 6 major functions in SPRS. The specifics of these components are detailed in Section 3.0, Requirements Specifications.



SEQUENCE DIAGRAM FOR PATIENT MANAGEMENT:-



SEQUENCE DIAGRAM FOR PATIENTBILLING MANAGEMENT:


Patient
Testing Bills
Medical Bills
Residing Bills
Total Bills
Enter Patient ID
Verify Tests undergone by Patient
Verify Medicines Undertaken by Patient
Calculate Bills for residing
Reply to Patient In Bill Form

System Feature ID: **System Feature Name:**

SRS_PBS _REG_01 *Patient* Registration

Description: Patient registration feature enables the registration process of the patient. The feature stores the data in a database and retrieve using registration ID. Updates the Patient data whenever necessary.

Activity Diagrams,

Sequence Diagrams,

Class Diagrams:

List the graphical models used to analyze the features and to better understand what needs to happen in the system. This can help identify issues such as omissions in the requirements. Remember that this is an analysis activity and not yet a design activity. Proceed only to a level of detail that allows you to understand the problem domain.

The diagrams are defined as delineated in the Unified Modeling Language (UML):

Activity diagrams that show the user-system interactions: Use this type of diagram to better understand the flow of the work, and, in particular, the branching.

Sequence diagrams that show object interactions: Use this type of diagram to identify the key concepts of the domain, i.e., the objects that will implement the behavior.

Class diagrams that summarize the classes of objects that appear in the sequence diagram: Use this type of diagram as a complement to the previous diagram.

System Feature ID: **System Feature Name:**

SRS_PBS_WARD_01 **WARD** Management

Description: This feature enable to maintain the wards(General/ICU) and about their
Patients information .

Activity Diagrams,

Sequence Diagrams,

Class Diagrams:

<List the graphical models used to analyze the features and to better understand what needs to happen in the system. This can help identify issues such as omissions in the requirements. Remember that this is an analysis activity and not yet a design activity. Proceed only to a level of detail that allows you to understand the problem domain.

The diagrams are defined as delineated in the Unified Modeling Language (UML):

Activity diagrams that show the user-system interactions: Use this type of diagram to better understand the flow of the work, and, in particular, the branching.

Sequence diagrams that show object interactions: Use this type of diagram to identify the key concepts of the domain, i.e., the objects that will implement the behavior.

Class diagrams that summarize the classes of objects that appear in the sequence diagram: Use this type of diagram as a complement to the previous diagram. >

System Feature ID:	**System Feature Name:**
SRS_PBS _BS_01	*BILLING SYSTEM*

Description: This feature enable to give billing information so that patient billls can be calculated easily.

Activity Diagrams, Sequence Diagrams, Class Diagrams: <List the graphical models used to analyze the features and to better understand what needs to happen in the system. This can help identify issues such as omissions in the requirements. Remember that this is an analysis activity and not yet a design activity. Proceed only to a level of detail that allows you to understand the problem domain.

The diagrams are defined as delineated in the Unified Modeling Language (UML):

Activity diagrams that show the user-system interactions: Use this type of diagram to better understand the flow of the work, and, in particular, the branching.

Sequence diagrams that show object interactions: Use this type of diagram to identify the key concepts of the domain, i.e., the objects that will implement the behavior.

Class diagrams that summarize the classes of objects that appear in the sequence diagram: Use this type of diagram as a complement to the previous diagram. >

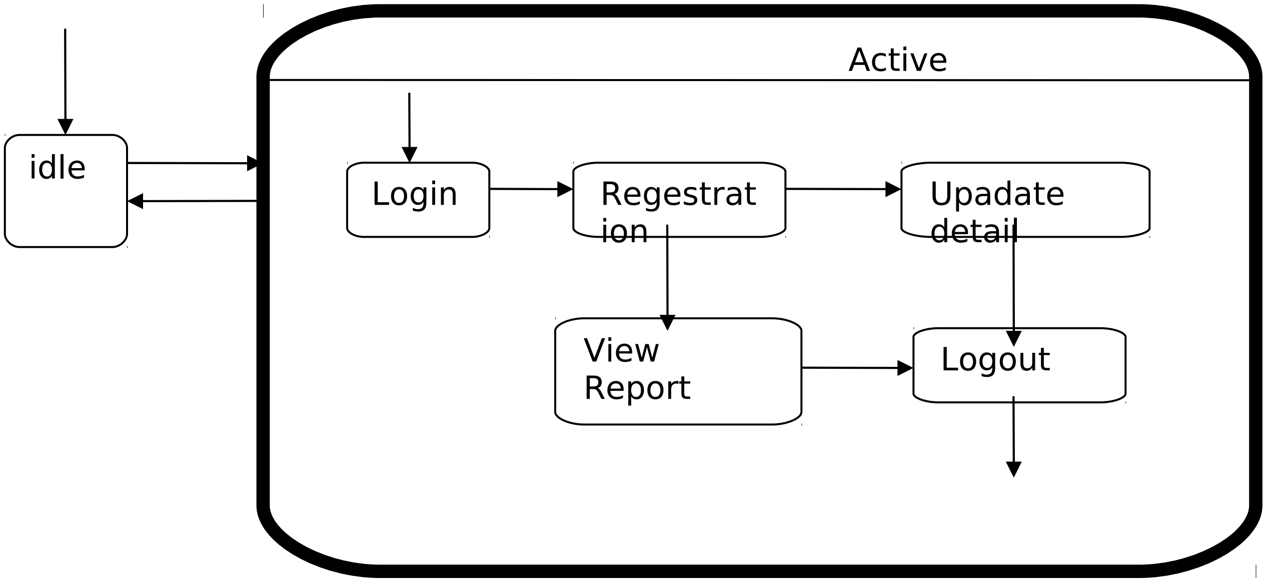
S. No.	**Class Name**	**Responsibility**	***Persistent (Y / N)***
S1	PATIENT	The patient must enter all the personal details	Permanent Database
S2	ADMINISTRATOR	Receptionist: He has the responsibilities to take the patient details and to assign to doctor concern for the disease suffering by the patient .	Permanent Database
S3	BILLS	Bills class has responsibility to maintain bills of each patient.	
S4	MEDICINE	This class has the responsibility to maintain the medicine details which are given to the patients..	
S5	DOCTOR DETAILS	This class has the responsibility to maintain Doctor‘s schedule ,and allotted the respective doctor to the patient.	
S6	WARD DETAILS	This class has the responsibility that should maintain the WARD	

details of patients and fee details of patient for General/ICU and also shifting of ward details.

State Analysis

Administrator (Receptionist) will be provided with a login name and password. When a valid user enters the system, a list of services will be displayed as hyper links. User can select any option and perform desired operations like master updating, deletions, insertions, taking reports etc. Data updating service will be provided to only designated user. Higher officials can see the reports. User can select logout option and exit from the system.

State Transition Diagram-



USER CHARACTERISTICS:-

The following agencies are expected to submit bills/personnel transactions to the PBS from internal billing and personnel systems:

End Users:

Hospital management officials, which include:

- Lab Authority
- Ward Authority
- Pharmacy Authority
- Administrator

Users have update access to PBS through batch processing and can perform online changes to PBS data. A limited number of inquiry screens are available to these agencies and Comptroller staff. The heaviest use of the data is among Lab Authority, Ward Authority, Pharmacy Authority, and Administrator

Only authorized people sign in to the system, others cannot sign in and update the daily transactions.

ASSUMPTIONS AND DEPENDENCIES:-

- The details related to the product, customer, payment and service transaction Provided manually.
- Administrator is created in the system already.
- Roles and tasks are predefined.
- Only authorized users can access the system.

FUNCTIONALITY:-

Functional Requirements:-

Feature Analysis	The PBS project module consists of following sub modules Admission of new patients. Search/view details of existing patients. • Transfer of patients. • Billing

System Feature ID: SRS_REG_01	**System Feature Name:** Patient Billing System.
Description:	Enables the hospital authorities to perform billing efficiently and maintain patient details without any discrepancies.
Activity Diagrams, Sequence Diagrams, Class Diagrams:	The diagrams are defined as delineated in the Unified Modeling Language (UML): Activity diagrams that show the user-system interactions: Use this type of diagram to better understand the flow of the work, and, in particular, the branching. Sequence diagrams that show object interactions: Use this type of diagram to identify the key concepts of the domain, i.e., the objects that will implement the behavior. Class diagrams that summarize the classes of objects that appear in the sequence diagram: Use this type of diagram as a complement to the previous diagram.
Operations:	Register new patients. Keep track of all the facilities used by the patients and charge them accordingly. Provide Bill to the patient.

RELIABILITY:-

The factors used to determine system reliability are:

- System should complete processing during the processing window.
- System shall provide a data check-point that saves all completed transactions.
- System shall retain historical records as follows:
  - System will archive to tape (tapes will be maintained for a period of seven years):
  - Patients’ transactions with an effective date prior to the current fiscal year plus two prior fiscal years.
  - Patients’ transactions with a payment date prior to the current calendar year plus two prior calendar years.

- System will archive to disc:

- Data extract files, until the next time the extract program is run. The System Input/Output Section within the Comptroller's office maintains 255 back-up versions of all data extract files transmitted to agencies.

- 255 back-up versions of all print extract files transmitted to agencies. After 255 versions are created, the System Input/Output Section moves them to tape and maintains them for an additional four years.

Safety and reliability Requirements	By incorporating a robust and proven RDBMS (MYSQL) into the system, reliable performance and integrity of data is ensured. There must be a power backup for server system

<u>PERFORMANCE</u>:-

PBS performs overnight batch processing. A batch queuing process ensures that batches are processed in the correct order. Batches are queued for processing by the date/time stamp of the batch on a first in, first out basis. Batches are submitted to the queue by the agency .

All PBS batches received before the specified time are processed that night. Batches received after the specified are processed the next night.

Performance Requirements	System can withstand even though all the authorities try to update the data in the database tables at the same time. Access is given to the only authenticated employees of a hospital who are assigned respective tasks.

<u>SECURITY</u>:-

<u>SYSTEM SECURITY</u>:-

PBS Security is managed through existing procedures within the Comptroller's office using system Security. A user identification (User ID) consisting of the users first name, users last name and the users' designation in hospital/department shall be assigned to all users accessing the system. In some cases, to avoid duplicate User IDs, the convention may vary. Only users' with a needed designation in hospital/department will be provided authorized logon privileges. Should an attempt to logon fail after three times, the User ID shall be locked out of the system until the user contacts a Security Coordinator to resolve the problem.

<u>DATA SECURITY</u>:-

SPRS will be backed up as part of the scheduled backups.

MAINTAINABILITY AND PORTABILITY:-

PBS is a mainframe system to be developed using software currently supported within the Comptroller's office.

INTERFACES:-

1. Login screen.
2. User id and Password screen
3. Update database details screen.
5. Delete database Details screen.
6. Billing screen.

HARDWARE INTERFACES:-

Server configuration:
Minimum 1GB Hard Disk
P-III processor or equivalent
RAM 128 MB
Windows or Linux

SOFTWARE INTERFACES:-

Operating System – Linux, Microsoft
Language -- visual basic6.0,
Database --Ms Access

COMMUNICATIONS INTERFACES:-

LAN protocols and TCP/IP
Mozilla web browser or equivalent

HIGH LEVEL DESIGN SPECIFICATION:-

PURPOSE

This document provides the detailed system design of Document Management System. It describes the high-level design that explains the functionality of the system. The purpose of this document defines the architecture, implementation, database design details of the system to the development team.

SCOPE

The Scope of this document includes the design details for the functioning of different modules of the system, database details are also included that is required for the storage of information.

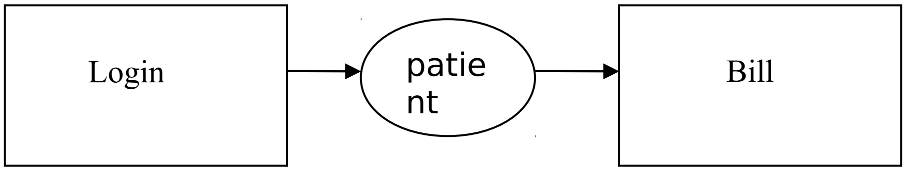
DESIGN CONSIDERATIONS

- The issues that measure the design are:
- User-friendly interface.
- Authorized users can only access the application.
- Access is limited by the authorization level.
- The new version replaces the older version when the document uploads
- Ms Access is the database manager and visual basics as a front end.

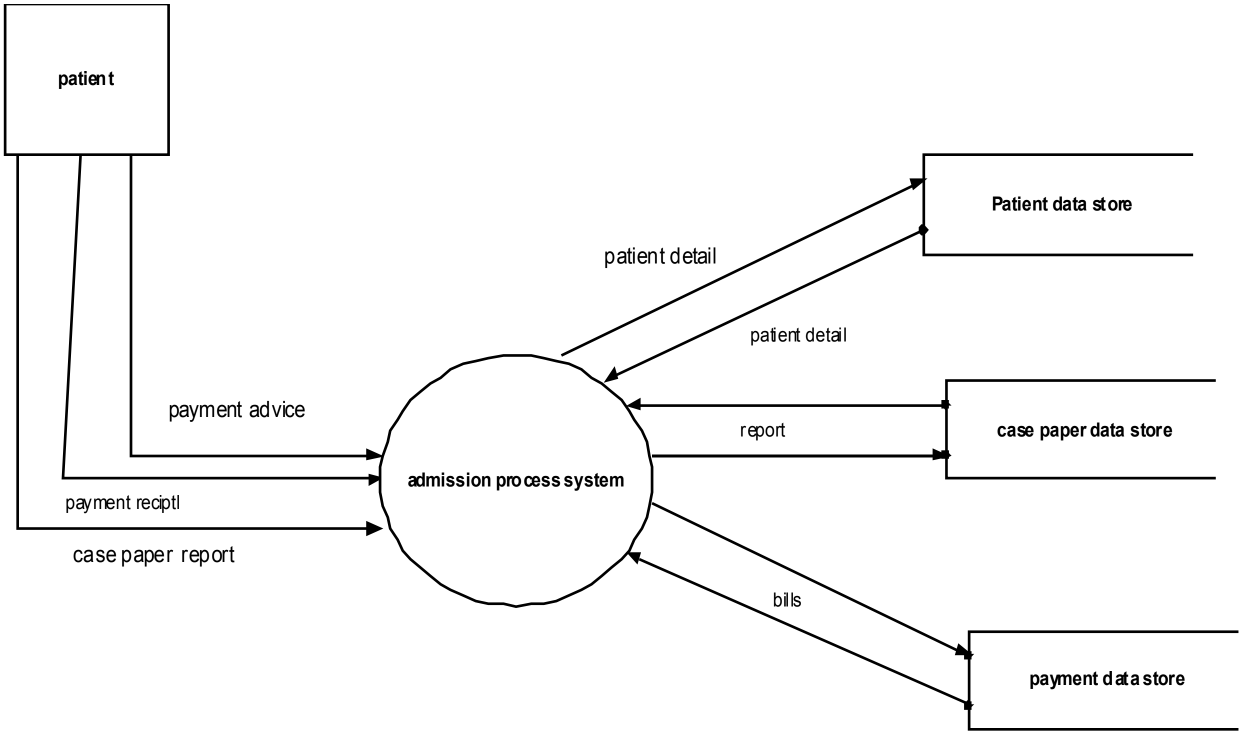
DESIGN:-

DFDs showing Patient Billing System-

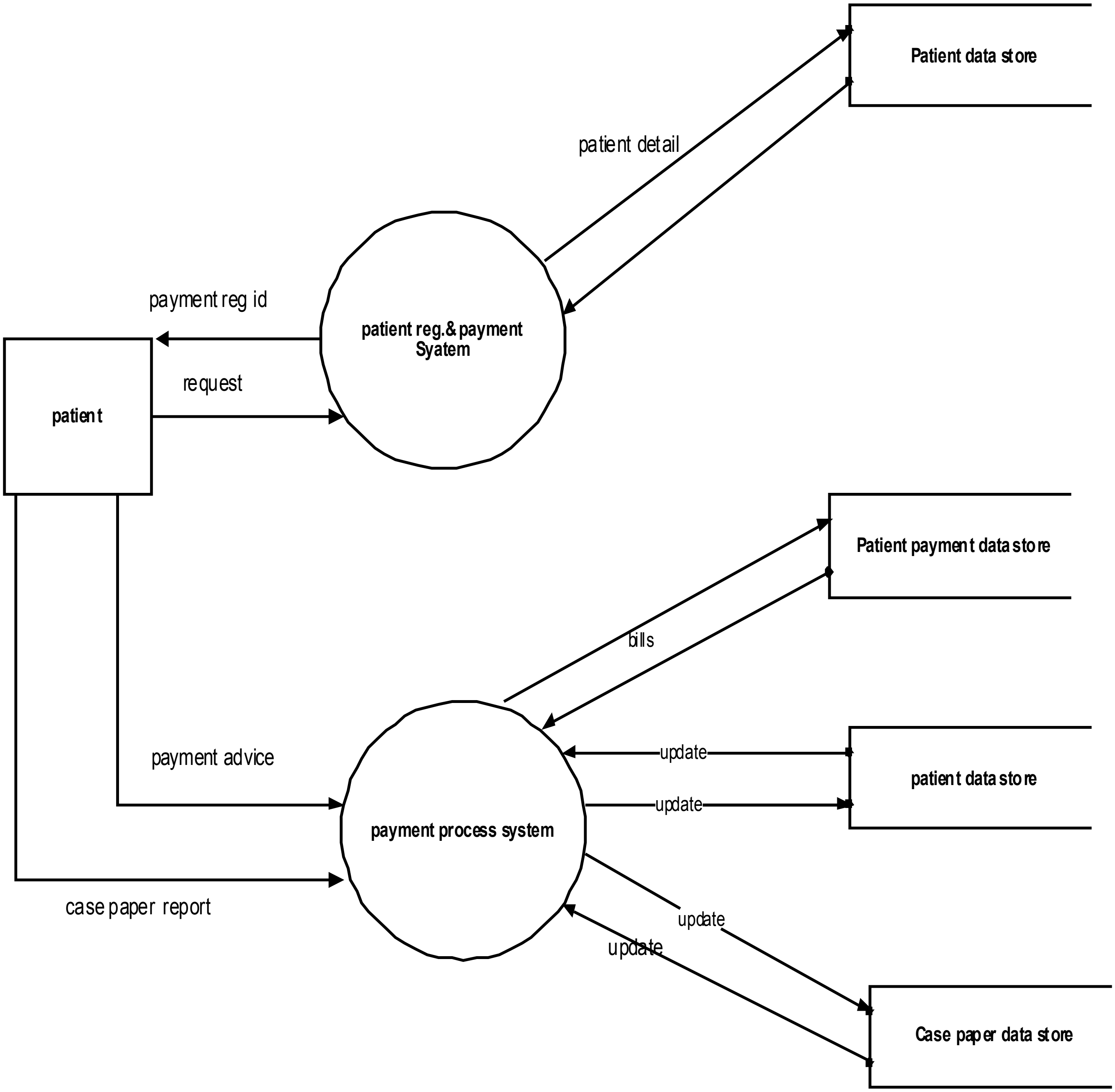
0 Level DFD -



1 Level DFD-


patien t
Patient data store
patient detail
patient detail
payment advice
admission process system
report
case paper data store
payment reciptl
case paper report
bills
payment d ata store

2 Level DFD-


Patient data store
patient detail
payment reg id
patient reg.& payment Syatem
request
patient
Patient payment data store
bills
update
patient data store
payment advice
update
payment process system
case paper report
update
update
Case paper data store

E-R DIAGRAM: -

Data models are tools used in analysis to describe the data requirements & Assumptions in the system from a top-down perspective .They also set stage for the design of databases.

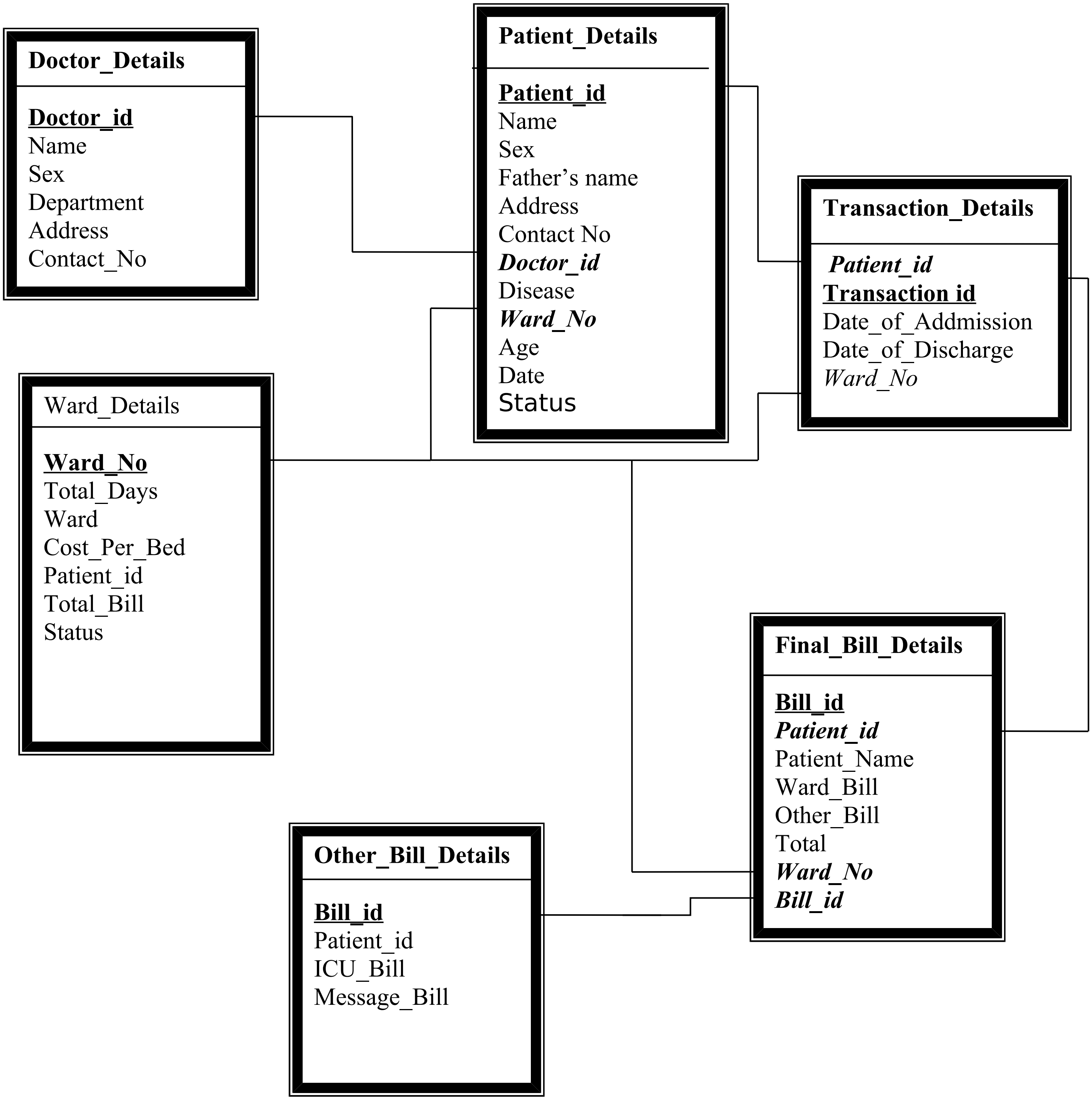
There are three basic elements in E-R Models:

- Entities are "things" about which we seek information
- Attributes are the data we collect about the entities.
- Relationships provide the structure needed to draw information from multiple entities.

DEVELOPING AN E-R DIAGRAM: -

Developing an E-R Diagram require an understanding of the system & its components before discussing the procedure, let's look at a narrative created by us.

RELATIONSHIP DIAGRAM



Doctor_Details

Doctor_id
Name
Sex
Department
Address
Contact_No

Patient_Details

Patient_id
Name
Sex
Father’s name
Address
Contact No
Doctor_id
Disease
Ward_No
Age
Date
Status

Transaction_Details

Patient_id
Transaction id
Date_of_Addmission
Date_of_Discharge
Ward_No

Ward_Details

Ward_No
Total_Days
Ward
Cost_Per_Bed
Patient_id
Total_Bill
Status

Final_Bill_Details

Bill_id
Patient_id
Patient_Name
Ward_Bill
Other_Bill
Total
Ward_No
Bill_id

Other_Bill_Details

Bill_id
Patient_id
ICU_Bill
Message_Bill

LOGICAL DATA ANALYSIS OF THE SYSTEM

DATA DICTIONARY: -

The classification of the Total Number, Types & Names of Databases & Tables used in the project are shown below here:

1.	Patient_id	Text
2.	Patient_Name	Text
3.	Sex	Text
4.	Father’s_Name	Text
5.	Address	Text
6.	Contact_No	Number
7.	Doctor_id	Text
8.	Disease	Text
9.	Ward_No	Number
10.	Age	Number
11.	Date	Date/Time
12.	Status	Text
13.	Doctor_Name	Text
14.	Department	Text
15.	Total_Days	Number
16.	Ward	Text
17.	Cost_Per_Bed	Currency
18.	Total_Bill	Currency
19.	Transaction	Text
20.	Date_of_Admit	Date/Time
21.	Date_of_Discharge	Date/Time
22.	ICU_Bill	Currency
23.	Medicine Bill	Currency
24.	Total	Currency
25.	Other_Bill	Currency
26.	User_Name	Text

27. Password	Text
28. Type	Text

DESIGN CONSTRAINTS:-

- GUI is only in English.
- Login and password is used for identification of users and there is no facility for guest.
- This system can be accessed by the authorized users only.
- There is no maintainability of back up so availability will get effected.

TEST PLAN:-

PURPOSE:-

This document is a high-level overview defining our testing strategy for the patient billing system. This document will address the different standards that will apply to the unit and functional testing of the application, to achieve correct code and ensure all functional and design requirements are implemented as specified in the SRS documentation.

The purpose of the Test Plan document is to:

- Specify the approach that Testing will use to test the application.
- Break the product down into distinct areas and identify features that are to be tested.
- Specify the procedures to be used for testing sign-off and product release.
- Indicate the tools used to test the product.
- List the resource and scheduling plans.
- Identify risks and contingency plans that may impact the testing.
- Specify bug management procedures for the project.

OBJECTIVE:-

The objective of our test plan is to find and report as many bugs as possible to improve the integrity of our program. Although exhaustive testing is not possible, we will exercise a broad range of tests to achieve our goal.

SCOPE:-

Testing will be conducted at the different level. The scope of testing will be limited to the top-level panels of the design. Both the functional aspect, i.e. validity of the reports generated, as well as compliance to the GUI guidelines will be verified, the look and feel of the screens, the overall functionality of the application. The Test Plan identifies the details of the test approach, identifying the associated test case areas within the specific product for this release cycle.

TESTING APPROACH:

Testing is the process of executing programs with the Intent of finding errors, rather than (a misconception) of showing the correct functioning of the programs. The distinction may sound like a matter of semantic, but it has been observed to have profound effect on testing success. Testing should bear the following objectives:

- to reveal design errors
- to reveal logic errors
- to reveal performance bottlenecks
- to reveal security loopholes
- to reveal operational deficiencies

The Test Approach sets the scope of system testing, the overall strategy to be adopted, the activities to be completed, the general resources required and the methods and processes to be used to test the release. It also details the activities, dependencies and effort required to conduct the System Test. The test approach should reveal the following details:

- New & revised Transaction Processing

- New Query Processes
- Revised Audit process
- Relocate Exceptions
- Revised Query Management process
- Revised Retrievals process

A successful testing approach is one that detects an undiscovered error.
A necessary part of a testing approach is a definition of the expected outputs or results. Do not plan testing effort on assumption that no errors will be found.
The probability of the existence of more errors in a section of a program is proportional to the number of errors already found in that section.
Testing libraries should be set up allowing Regression test to be performed at the time of system maintenance and enhancement. The later in the development life cycle a fault is discovered, the higher is the cost of correction. Successful testing relies on complete and unambiguous specification.

TEST PHASES:-

Test Phases	Testing Strategy Applied

Unit Testing **-** Testing of the program modules in isolation with the objective to find discrepancy between the programs and the program specifications.	White box testing in LLD
Integration Testing **-** Testing of the linkages between testing program modules with the objective to find discrepancy between the programs and system specifications.	White box testing in LLD
Function Testing **-** Testing of the integrated software on a function-by-function basis with the objective to find discrepancy between the programs and the function specifications.	Black box testing
Systems Testing **-** Testing of the integrated software with the objective to find discrepancy between the programs and the original objectives with regard to the operating environment of the system (e.g. Recovery, Security, Performance, Storage, etc.).	Black box testing
Acceptance Testing **-** Testing of the integrated software by the end-users (or their proxy) with the objective to find discrepancy between the programs and the end-user needs.	Black box testing

SCOPE OF TESTING:-

System testing is the process of testing the integrated software with regard to the operating environment of the system.(i.e. Recovery, Security, Performance, storage, etc)

It's worthwhile to note that the term has been used with different environments. In its widest definition especially for the small scale projects, it also covers the scope of integration testing and the function testing.

For our application which combines the integration testing, functional testing, system testing in one set.

All the testing done at the customers site are hopefully well and are similar to those are shown in the description of design phase of these software. The entire input screens are same. All output screen are same ,because we used the technique provided by MICROSOFT in our program so that if a computer can satisfy a minimum set of hardware and software requirement then out program behave equally well on both computer that on development site or customer's site.

Therefore if you want to see any data, input screen, output screen reports and printouts then you can see them in the previous description of design phase.

ENTRY CRITERIA:-

The Entrance Criteria specified by the system test controller, should be fulfilled before System Test can commence. In the event, that any criterion has not been achieved, the System Test may commence if Business Team and Test Controller are in full agreement that the risk is manageable.

All developed code must be unit tested. Unit and Link Testing must be completed and signed off by development team.

System Test plans must be signed off by Business Analyst and Test Controller.

All human resources must be assigned and in place.

All test hardware and environments must be in place, and free for System test use.

EXIT CRITERIA:-

All High Priority errors from System Test must be fixed and tested

If any medium or low-priority errors are outstanding - the implementation risk must be signed off as acceptable by Business Analyst and Business Expert

Project Integration Test must be signed off by Test Controller and Business Analyst.

Business Acceptance Test must be signed off by Business Expert.

REQUIREMENT OF SYSTEM TESTING:-

Testing is vital to the system. System testing makes a logical assumption that if all the parts of the system are correct, the goal be successfully achieved. Contesting leads to errors that create problems.

The time lag between the cause and the appearance of the problem The effect of system errors on files and records within the system. A small system error can conceivably explode into a much problem. Effective testing early the process translate directly into term cost saving from a reduced number of error.

System testing is usable as a user oriented vehicle before implementation. It is helpful to a system tester to bridge the communicate barrier.

LIMITATION OF THE PROJECT:-

This project will not work for all hospitals. The project I am making is a real life project. Still there are few loopholes, which should be taken care in the long term. The project I am making is used in fewer organizations because it is a traditional system of maintaining the patient billing management. So as and when time changes few more details may be required in this project, hence we have update it at regular interval of time. Few limitations that I can oversee at this point of time are as under:-

- This application is for standalone machine.
- Typical (or as told by the clients).

BIBLIOGRAPHY

References

Steven Holzner, Visual Basic 6 Programming Black Book, Dream Tech Press, 2007

Evangelos Petroutsos, Mastering Visual Basic 6, BPB Publication, 2006

Rajib Mall, Fundamentals of Software Engineering, PHI Learning Private Limited, 2008

Sajan Mathew, Software Engineering, S.Chand & Company Ltd, 2007

Prabhakar Gupta, Software Engineering, Pragati Prakashan, 2007

Roger S Pressman, Software Engineering a Practitioner's Approach, Mcgraw Hill International Edition, 2005

Zultner R, Quality Function Deployment for software satisfying customers, American Programmer, 1992

Gilb T, Principles of Software Engineering Management, Addison Wesley, 1998

Budd T, fundamental of data base management, Addison-Wesley, 1996