***An industry oriented mini project report submitted***

***In partial fulfillment of the requirement for the award of the degree of***

**BACHELOR OF TECHNOLOGY**

**IN**

**COMPUTER SCIENCE AND ENGINEERING**

*by*

**A.Navya Renuka(Reg. No: 14131A0504)**

**Ch.Vikas Sai(Reg. No: 14131A0519)**

**G.Sai Praneeth (Reg. No: 14131A0540)**

*Under the esteemed guidance of*

**Mr. Ch. AVINASH**

**Assistant Professor**

**Department of Computer Science and Engineering**



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

GAYATRI VIDYA PARISHAD COLLEGE OF ENGINEERING(A)

(AFFILIATED TO JNTU, KAKINADA, AP)

VISAKHAPATNAM – 530048

2014 – 2018



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

GAYATRI VIDYA PARISHAD COLLEGE OF ENGINEERING(A)

(AFFILIATED TO JNTU, KAKINADA, AP)

VISAKHAPATNAM – 530048

**CERTIFICATE**

This is to certify that the project work entitled “**ONLINE MEDICAL ADVISOR**” being submitted by A.Navya Renuka(14131A0504), Ch.Vikas Sai(14131A0519),

G.Sai Praneeth(14131A0540) in partial fulfillment of the requirement the award of the degree of “Bachelor of technology” in Computer Science and Engineering is a record of bonafide work done by them under my supervision during the academic year 2016.

**Internal guide Head of the Department**

**Mr.Ch.Avinash Dr.P.Krishna Subba Rao Assistant ProfessorProfessor & Head of Department CSE**

**GVPCOE(A) GVPCOE(A)**

**DECLARATION**

We hereby declare that this is dissertation of our own work except where specifically ask to the contrary and it is not substantially the same as any dissertation which has been submitted to any university.

**By**

**A.Navya Renuka(14131A0504)**

**Ch.Vikas Sai(14131A0519)**

**G.Sai Praneeth(14131A0540)**

**ACKNOWLEDGEMENT**

We thank **Prof. A.B.K.Rao** principal, **Gayatri Vidya Parishad College of Engineering(A)**  for extending his utmost support and cooperation in providing all the provisions for the successful completion of the project.

We consider it our privilege to express our deepest gratitude to **Dr.P.K.Subba Rao** Professor, Head of the Department, Computer Science and Engineering, for his valuable suggestions and constant motivation that greatly helped the project work to get successfully completed.

We thank our guide **Mrs.Hindhumati** Assistant Professor, Computer Science and Engineering, for suggesting and guiding us in completing our project successfully.

We also thank all the members of the staff in Computer Science Engineering for their sustained help in our pursuits.

We thank all those who contributed directly or indirectly in successfully carrying out his work.

**By**

**A.Navya Renuka(14131A0504)**

**Ch.Vikas Sai(14131A0519)**

**G.Sai Praneeth(14131A0540)**

**ONLINE**

**MEDICAL**

**ADVISOR**

**ABSTRACT**

The aim of the project is to prepare schedules for ringing of a alarm bell, on a particular day. Project provides great feasibility for managing the schedules such as creating , editing and removing. Until recent times ringing of bell in colleges and schools is manual, man work plays a vital role. We took it as a challenge and made the procedure automated. User can prepare schedules on a particular day in the future and manage it in further days. The working of a particular schedule can be started as soon as the server starts, which sends the signal for the alarm to ring when the time assigned in the schedule matches with current running time. Taking the exacerbate case of power cut, the schedule starts from where it stopped.

**INDEX**

pageno

1. ABOUT THE PROJECT…………………………………………...... 12

2. PROJECT SCOPE…………………………………………………… 13

2.1 EXISTING SYSTEM………………………………………….. 14

2.2 PROPOSED SYSTEM………………………………………… 14

3. FEASIBILTY REPORT……………………………………………… 15

3.1 TECHNICAL DESCRIPTION………………………………… 16

3.2 NUMBER OF MODULES……………………………………… 17

3.3 REQUIRED HARDWARE……………………………………… 18

3.4 REQUIRED SOFTWARE……………………………………… 18

3.5 FEASIBILITY TYPES………………………………………… 19

4. ANALYSIS…………………………………………………………… 20

4.1 SRS DOCUMENT……………………………………………… 21

4.2 SCOPE OF DEVELOPMENT………………………………… 22

4.3 ABOUT ANDROID…………………………………………… 23

4.4EXTENSIBLE MARKUP LANGUAGE ……………………… 28

4.5 CONNECTIVITY ……………………………………………… 32

5. DESIGN…………………………………………………………… 34

5.1 UML SPECIFICATIONS…………………………………………… 37

5.2 SEQUENCE DIAGRAM…………………………………………… 38

5.3 CLASS DIAGRAM………………………………………………… 39

5.4 ACTIVITY DIAGRAM……………………………………………. 40

6. CODING……………………………………………………………… 41

6.1 SAMPLE CODE…………………………………………………….. 42

6.2 OUTPUT SCREENS………………………………………………… 60

7. TESTING………………………………………………………………. 63

7.1 LEVELS OF TESTING……………………………………………… 65

8. CONCLUSION………………………………………………………… 66

9. BIBLIOGRAPHY ……………………………………………………… 68

9.1 REFERENCES……………………………………………………… 69

9.2 LIST OF WEBSITES………………………………………………… 69

1. **ABOUT THE PROJECT**

Admin gets a request URL for running the current day schedule, if no schedule is present, he can create a one on that instant. Any current changes in the previous schedule of that day can be made. Lastly he runs the schedule. Moving to the managing part which is kernel of the project, the working code of the project is written in javascript, running over NODE.JS framework, which provides greater flexibility for managing the work. Admin will be given a managing URL for creating schedules; he can select a particular day for creating/editing/deleting a schedule. All the schedules are stored in MYSQL database. Since the managing part hovers over mapping a schedule for a particular date, relational database is used. Signal from server is given to serial port, from there to 8051 micro controller.

**2. PROJECT SCOPE**

**2.1 Existing System:**

Current system for ringing of a bell at particular time is manual, man work is included .Schedule at which the bell has to ring is notified to a person who’s job is to ring the bell manually. Many of the colleges have updated the system digitally and making the bell to ring by sending a signal but, this system of working is good when all the days have to run on same schedule and storing of future days schedule is impossible. These type of systems have no scalability, feasibility. Accuracy of the time at which bell rings depends on the man work.

It’s a huge headache for the management to control schedule for the future days and manage them. A prefixed schedule for ringing of a bell exists in the android level but, android alone is not sufficient for making it scalable for a management system like college or school. Existig system doesn’t make use of technologies for making a better management for ringing of a bell

**2.2. Problems in Existing system :**

* Mismatching of the bell ringing time and original time at which it is expected to ring.
* Saving of the future day schedules.
* Innumerable timings for a particular schedule are not present.

**Proposed System**

We have provided many advantages to the users of our site. The data related to the user is confidential as we have provided security for each and every user. They have to register and then login with the password in order to access the site. In this way, the info of one user will be not known to others. The number of symptoms to choose is not limited. For eg, if we have five symptoms we can choose all the five. And finally we will give the prescription to the user and asks for the rating to know how much he is satisfied with the prescription given. In addition we will give some medical tips to the users of the site. We realize the importance of maintaining up-to-date and accessible medical information on their websites.So we will update the information on the site from time to time with the current high quality medicines which are in use. The information provided by our site is reliable and trustworthy as we have consulted many people and taken care of the medicine to be prescribed in a right time in a proper way from which patients can maximally benefit from the information opportunities afforded by the internet.

**3. Feasibility Report**

**3.1 Technical Description:**

**Layouts:**

**Eclipse**:

**GUI’s**

**1) The Android Emulator:**

**2)Mobile device**:

**Databases:**

**Shared Preferences:**

**3.2 MODULES**

The system after careful analysis has been identified to present with the following modules.

1. **Current Day Module:**
2. **Managing module:**

**3.3 REQUIRED HARDWARE :**

Hardware requirements

* System : GENERAL PC, HAVING A SERIAL PORT OF 15PIN/9PIN
* Serial port(COMX)
* Monitor : FOR VIEWING THE RESULT OF SCHEDULES
* Serial port to USB port converter: FOR TRANSMITTING THE DATA FROM SERIAL PORT TO USB PORT

**3.4 REQUIRED SOFTWARE:**

Software requirements

* Operating system : WINDOWS
* Coding Language : JAVASCRIPT
* EDITOR : ATOM

**3.5 FEASIBILITY TYPES:**

**Technical feasibility:**

The technical issue usually raised during the feasibility stage of the investigation includes the following:

* Does the necessary technology exist to do what is suggested?

All the technologies that we proposed to use are free and open source, so they are easily available and obtainable for almost any software platform. Therefore the possession of the technology is not a problem.

* Do the proposed equipments have the technical capacity to hold the data required to use the new system?

All the software requirements are platform dependent, but every software requirement is open source and would be available on cloud.

* Does the project run on various operating systems?

Of course, it runs on various operating systems. All the software needed is node.js and atom, which are open source and MYSQL can be easily installed and sql copy of the existing database can be downloaded and imported into MYSQL on new platform.

.

**Financial Feasibility:**

As the project is confined to a college or school , admin will have to invest money for buying a 8051 controller which is less in cost. If the admin want to handle the schedules on cloud he should invest in maintaining a database in the cloud which could be integrated to the project.

**4. ANALYSIS Report**

**4.1 SRS DOCUMENT**

**Intended Audience and Reading Suggestions**

The document is prepared keeping is view of the academic constructs of my Bachelors Degree / Masters Degree from university as partial fulfillment of my academic purpose the document specifies the general procedure that that has been followed by me, while the system was studied and developed. The general document was provided by the industry as a reference guide to understand my responsibilities in developing the system, with respect to the requirements that have been pin pointed to get the exact structure of the system as stated by the actual client.

The system as stated by my project leader the actual standards of the specification were desired by conducting a series of interviews and questionnaires. The collected information was organized to form the specification document and then was modeled to suite the standards of the system as intended.

**Document Conventions:**

The overall documents for this project use the recognized modeling standards at the software industries level.

* + The Physical dispense, which state the overall data search for the relational key whereas a transaction is implemented on the wear entities.
  + Unified modeling language concepts to give a generalized blue print for the overall system.
  + The standards of flow charts at the required states that are the functionality of the operations need more concentration.

**4.2 SCOPE OF DEVELOPMENT**

Firstly, the project is scalable to larger extent which makes it to be developed to higher levels by integrating with artificial intelligence. The database could be maintained in the cloud and make the same schedule to run in different colleges.

**Future scope:**

* One database connection to many systems allows many organizations to work on same schedule.
* Examinations which are held commonly at the same time in various places will have a bell ring at the same time.
* Probability of mismatching bell rings during work will is reduced.

**4.3 ABOUT ANDROID :**

**Introduction to Android**

#### The Android Architecture

.

#### Android Developer Tool (ADT)

#### Programming Tools:

**4.4 EXTENSIBLE MARKUP LANGUAGE**

#### 4.5 CONNECTIVITY

**5. Design Document**

**5.1 Unified Modeling Language Specifications**

This specification defines the Unified Modeling Language (UML), revision 2. The objective of UML is to provide system architects, software engineers, and software developers with tools for analysis, design, and implementation of software-based systems as well as for modeling business and similar processes. Relative to UML 1, this revision of UML has been enhanced with significantly more precise definitions of its abstract syntax rules and semantics, a more modular language structure, and a greatly improved capability for modeling large-scale systems.

One of the primary goals of UML is to advance the state of the industry by enabling object visual modeling tool interoperability. However, to enable meaningful exchange of model information between tools, agreement on semantics and syntax is required. UML meets the following requirements:

• A formal definition of a common MOF-based metamodel that specifies the abstract syntax of the UML. The abstract syntax defines the set of UML modeling concepts, their attributes and their relationships, as well as the rules for combining these concepts to construct partial or complete UML models.

• A detailed explanation of the semantics of each UML modeling concept. The semantics define, in a technology-independent manner, how the UML concepts are to be realized by computers.

• A specification of the human-readable notation elements for representing the individual UML modeling concepts as well as rules for combining them into a variety of different diagram types corresponding to different aspects of modeled systems.

We need to consider the primary modelling purposes of UML. These are:

* Business Process Modelling with Use Cases
* Class and Object Modelling
* Behaviour Modelling
* Component Modelling
* Distribution and Deployment Modelling

Each UML model is designed to let analysts, developers and customers view a system

from different perspectives and with varying levels of abstraction. Each diagram will fit

somewhere into these five architectural views representing a distinct problem solution space.

These can be described as the user model view, structural model view, behavioural model view

implementation model view and the environment model view.

**User Model View**

The UML user model view encompasses the models which define a solution to a problem

as understood by the client or stakeholders. This view is often also referred to as the Use Case or

scenario view. The main UML model encompassed by this view is the:

* **Use Case Diagram:** These models depict the functionality required by the system and the

interaction of users and other elements (known as actors) with respect to the specific

solution.

**Structural model view**

The UML structural view encompasses the models which provide the static, structural

dimensions and properties of the modelled system. This view is often also referred to as the

static or logical view. UML Models applicable to this view include:

* **Class Diagrams:**These models describe the static structure and contents of a system using

elements such as classes, packages and objects to display relationships such as

containment, inheritance and associations.

* **Object Diagrams:** Depict a class or the static structure of a system at a particular point in

time.

**Behavioral Model View**

These UML models describe the behavioural and dynamic features and methods of the modelled

system. This view is often also referred to as the dynamic, process, concurrent or collaborative

view. UML Models applicable to this view include:

* **Sequence Diagram:** Describe timing sequence of the objects over a vertical time

dimension with interactions between objects depicted on a horizontal dimension.

* **Collaboration Diagrams:** Describe the interactions and relationships between objects and

sequences of a system organised in time and space. Numbers are used to show the sequence

of messages.

* **State Diagrams:** Describe the sequence, status conditions and appropriate responses or

actions to conditions during the life of the objects within the system.

* **Activity Diagrams:** Describe the methods, activities and resulting transitions after

completion of the elements as flows of processing within a system.

**Implementation Model View**

The UML Implementation View combines the structural and behavioural dimensions of

the solution realisation or implementation. The view is often also referred to as the component

or development view. UML Models applicable to this view include:

* **Component Diagrams:** These depict the high level organisation and dependencies of

source code components, binary components and executable components and whether these

components exist at compile, link or run time.

**Environmental Model View**

These UML models describe both structural and behavioural dimensions of the domain or

environment in which the solution is implemented. This view is often also referred to as the

deployment or physical view. UML Models applicable to this view include:

* **Deployment Diagrams:** These UML Models depict and describe the environmental

elements and configuration of runtime processing components, libraries and objects that

will reside on them.

**UML:**

UML is a standard language for specifying, visualizing, constructing, and documenting

the artifacts of software systems.UML was created by the Object Management Group (OMG)

and UML 1.0 specification draft was proposed to the OMG in January 1997.OMG is

continuously making efforts to create a truly industry standard.

* UML stands for **Unified Modelling Language**.
* UML is different from the other common programming languages such as C++, Java,

COBOL, etc.

* UML is a pictorial language used to make software blueprints.
* UML can be described as a general purpose visual modelling language to visualize,

specify, construct, and document software system.

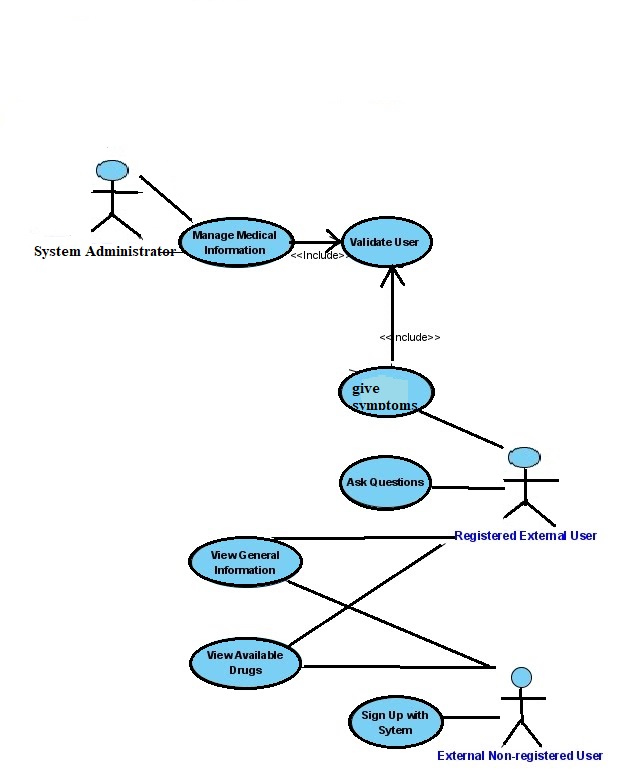
* Although UML is generally used to model software systems, it is not limited within this

boundary. It is also used to model non-software systems as well. For example, the

process flow in a manufacturing unit, etc.

**5.2.1. USE CASE DIAGRAM:**

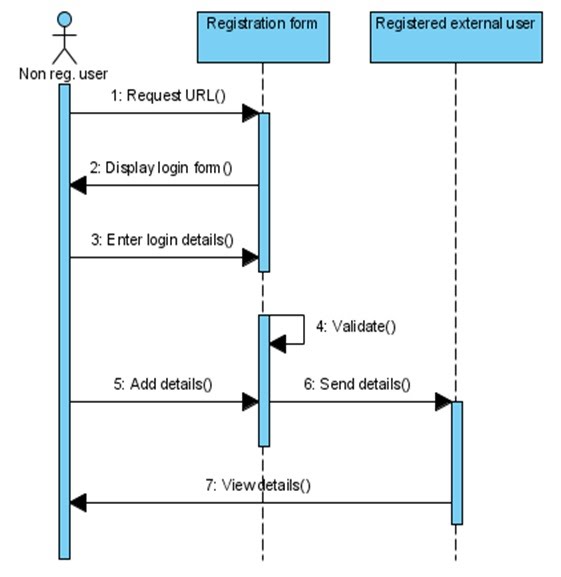
“A use case in software engineering and systems engineering is a description of a system’s behavior as it responds to a request that originates from outside of that system. In other words, a use case describes "who" can do "what" with the system in question. The use case technique is used to capture a system's behavioral requirements by detailing scenario driven threads through the functional requirements.”

****

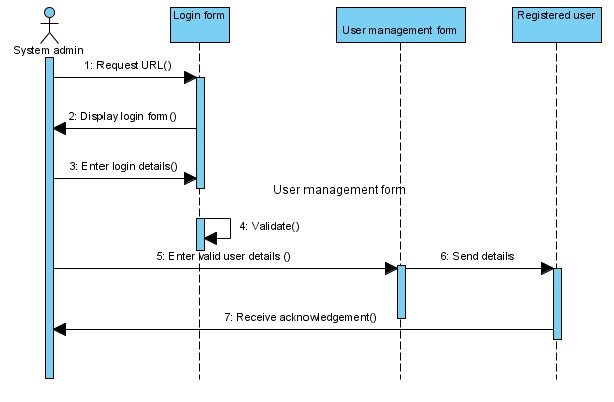
**5.2.2. SEQUENCE DIAGRAM:**

Sequence diagrams are used to capture the order of messages flowing from one object to another.

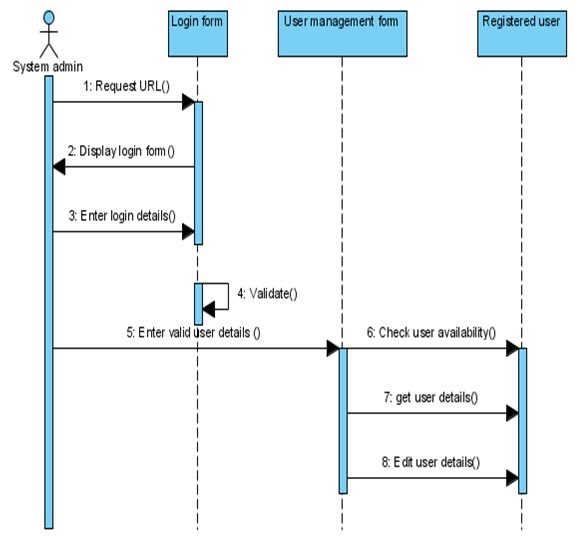
**USER REGISTRATION:**



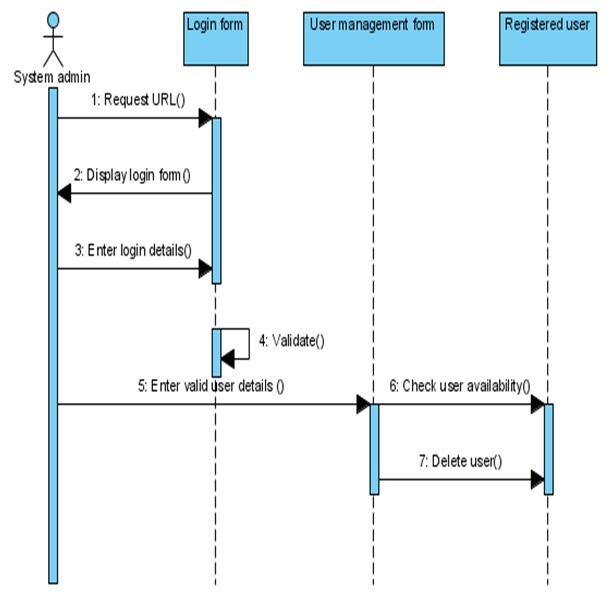
**ADD NEW USERS TO THE SYSTEM:**

****

**UPDATE USER DETAILS:**

****

**DELETE USER:**

****

**5.2.3 CLASS DIAGRAM :**

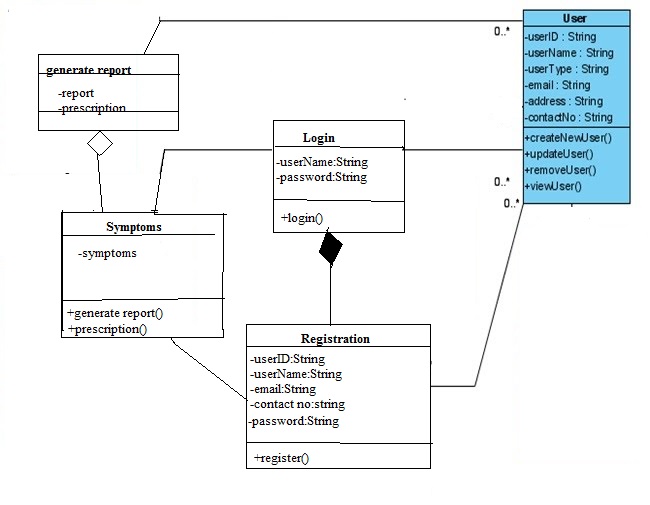
Class diagram is a static diagram. It represents the static view of an application. Class diagram is not only used for visualizing, describing, and documenting different aspects of a system but also for constructing executable code of the software application.

Class diagram describes the attributes and operations of a class and also the constraints imposed on the system. The class diagrams are widely used in the modelling of object oriented systems because they are the only UML diagrams, which can be mapped directly with object-oriented languages.

Class diagram shows a collection of classes, interfaces, associations, collaborations, and constraints. It is also known as a structural diagram.

The purpose of the class diagram can be summarized as −

* Analysis and design of the static view of an application.
* Describe responsibilities of a system.
* Base for component and deployment diagrams.
* Forward and reverse engineering.



**5.2.4 ACTIVITY DIAGRAM:**

Activity diagram is another important diagram in UML to describe the dynamic aspects of the system.

Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system.

The control flow is drawn from one operation to another. This flow can be sequential, branched, or concurrent. Activity diagrams deal with all type of flow control by using different elements such as fork, join, etc.

The basic purposes of activity diagrams is similar to other four diagrams. It captures the dynamic behavior of the system. Other four diagrams are used to show the message flow from one object to another but activity diagram is used to show message flow from one activity to another.

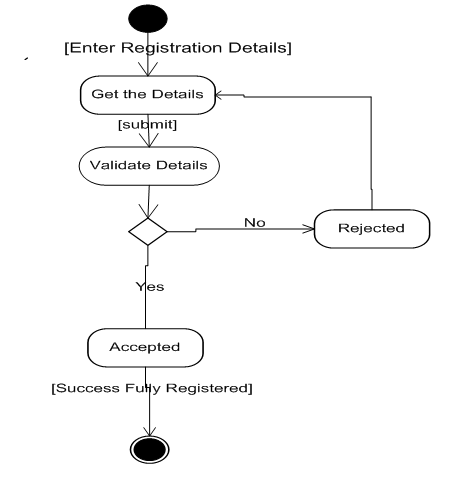
Activity is a particular operation of the system. Activity diagrams are not only used for visualizing the dynamic nature of a system, but they are also used to construct the executable system by using forward and reverse engineering techniques. The only missing thing in the activity diagram is the message part.

It does not show any message flow from one activity to another. Activity diagram is sometimes considered as the flowchart. Although the diagrams look like a flowchart, they are not. It shows different flows such as parallel, branched, concurrent, and single.

The purpose of an activity diagram can be described as −

* Draw the activity flow of a system.
* Describe the sequence from one activity to another.
* Describe the parallel, branched and concurrent flow of the system.

**REGISTRATION:**

****

**6. CODING**

**MainActivity.java :**

**6.2 Output Screen :**

**7.TESTING**

**Testing :**

* Testing is the process of evaluating a system or its component(s) with the intent to find whether it satisfies the specified requirements or not.
* Testing is executing a system in order to identify any gaps, errors, or missing requirements in contrary to the actual requirements.
* According to ANSI/IEEE 1059 standard, Testing can be defined as - A process of analyzing a software item to detect the differences between existing and required conditions (that is defects/errors/bugs) and to evaluate the features of the software item.
* It depends on the process and the associated stakeholders of the project(s).
* In the IT industry, large companies have a team with responsibilities to evaluate the developed software in context of the given requirements.

In most cases, the following professionals are involved in testing a system within their respective capacities:

* Software Tester
* Software Developer
* Project Lead/Manager
* End User

An early start to testing reduces the cost and time to rework and produce error-free software that is delivered to the client. However in Software Development Life Cycle (SDLC), testing can be started from the Requirements.

Testing is done in different forms at every phase of SDLC:

* During the requirement gathering phase, the analysis and verification of requirements are also considered as testing.
* Reviewing the design in the design phase with the intent to improve the design is also considered as testing.

**Purpose of Testing:**

**Testing** is the most time consuming, but a needful activity of a software project. It is vital

to the success of new system.  
  
The main **Purpose of testing** is to:

1. Measure the quality of the software
2. Find out the faults in the application and software
3. Reduce the number of bugs in the program.
4. Check all the requirements and specifications given by client and customers are met or not.
5. Produce a product which is full of quality oriented.
6. Comfort the need of client and customers.
7. Provide a defect and error free software.
8. Check whether the application is working as per the functional requirement specified or not.
9. Carry the difference between the expected and actual results.

Importance of  **purpose of testing** is to increase the project and product quality, and

definitely to increase the organization quality side by side also. At end we conclude that

during **software testing**, the test engineer needs to discover as several bugs as possible before

reaching those defects to client, which will definitely save lots of time and money.

Testing Objectives:

Software Testing  has different goals and objectives. The major objectives of Software

testing are as follows:

* [**Finding defects**](http://istqbexamcertification.com/what-is-defect-or-bugs-or-faults-in-software-testing/) which may get created by the programmer while developing the

software.

* Gaining confidence in and providing information about the level of [**quality**](http://istqbexamcertification.com/what-is-software-quality/).
* To prevent defects.
* To make sure that the end result meets the business and user requirements.
* To ensure that it satisfies the BRS that is Business Requirement Specification and SRS

that is System Requirement Specifications.

* To gain the confidence of the customers by providing them a quality product.

Software testing helps in finalizing the software application or product against business

and user requirements. It is very important to have good test coverage in order to test the

software application completely and make it sure that it’s performing well and as per the

specifications.

Once the delivery is made to the end users or the customers they should be able to

operate it without any complaints. In order to make this happen the tester should know as how

the customers are going to use this product and accordingly they should write down the test

scenarios and design the test cases. This will help a lot in fulfilling all the customer’s

requirements.

**7.1 Levels of Testing**

In order to uncover the errors present in different phases we have the concept of levels of testing. The basic levels of testing are as shown below…

System Testing

System testing tests the system as a whole. Once all the components are integrated, the

application as a whole is tested rigorously to see that it meets the specified Quality Standards.

This type of testing is performed by a specialized testing team.

System testing is important because of the following reasons:

* System testing is the first step in the Software Development Life Cycle, where the application is tested as a whole.
* The application is tested thoroughly to verify that it meets the functional and technical specifications.
* The application is tested in an environment that is very close to the production environment where the application will be deployed.
* System testing enables us to test, verify, and validate both the business requirements as well as the application architecture.

Code Testing

Code-based testing corresponds to the testing that is carried out on code development, code inspection, unit testing in software development process.

The Code-based testing consists of following testing:

* Dynamic Testing - Statement coverage, Branch coverage, Path coverage
* Checking for Complexity of Code using techniques like Cyclomatic Complexit
* Static Testing - Code Inspection, Code Walkthrough, Code Review, Code Audit

White Box Testing

White box testing is a testing technique, that examines the program structure and derives test data from the program logic/code. The other names of glass box testing are clear box testing, open box testing, logic driven testing or path driven testing or structural testing.

## White Box Testing Techniques:

* **Statement Coverage -** This technique is aimed at exercising all programming statements with minimal tests.
* **Branch Coverage -**This technique is running a series of tests to ensure that all branches are tested at least once.
* **Path Coverage -**This technique corresponds to testing all possible paths which means that each statement and branch is covered.

## Advantages of White Box Testing:

* Forces test developer to reason carefully about implementation.
* Reveals errors in "hidden" code.
* Spots the Dead Code or other issues with respect to best programming practices.

## Disadvantages of White Box Testing:

* Expensive as one has to spend both time and money to perform white box testing.
* Every possibility that few lines of code are missed accidentally.
* In-depth knowledge about the programming language is necessary to perform white box testing.

White Box Testing method is applicable to the following levels of software testing:

* [Unit Testing](http://softwaretestingfundamentals.com/unit-testing/): For testing paths within a unit.
* [Integration Testing](http://softwaretestingfundamentals.com/integration-testing/): For testing paths between units.
* [System Testing](http://softwaretestingfundamentals.com/system-testing/): For testing paths between subsystems.

However, it is mainly applied to Unit Testing.

Black Box Testing

Black-box testing is a method of software testing that examines the functionality of an application based on the specifications. It is also known as Specifications based testing. Independent Testing Team usually performs this type of testing during the software testing life cycle.

This method of test can be applied to each and every level of software testing such as unit, integration, system and acceptance testing.

## Black box Testing Techniques:

There are different techniques involved in Black Box testing.

* Equivalence Class
* Boundary Value Analysis
* Domain Tests
* Orthogonal Arrays
* Decision Tables
* State Models
* Exploratory Testing
* All-pairs testing

**8.CONCLUSION**

The proposed Online Medical Advisor system is basically a website that provides a lot of useful information to the general public on a wide variety of health and health related issues. Such as, detailed report, detailed prescription are given as per the user requirements.

Most importantly the system would contain information from both the indigenous & western medical Systems. Already there are medical information sources available on the internet. But, none of these provides information from both the medical systems. So, that users can effectively compare and decide the best treatment for them out of a wider variety of choices.

The site will also facilitate its users to direct their health related questions to medical experts. Interesting health tips to keep up a healthy life will also be available on the site.

Benefits through our site will contain,

* Creating public awareness of diagnosis and their general treatment.
* Redirecting them to medical professionals for any other queries (if they want know more).
* User should register first to log on to the site. So every time he need to log in and therefore he can view his previous health issues if any.
* Generating an over all report of user when he needs.
* Updating the information from time to time.

This is only for small health issues for which we can aid ourselves from home itself. But for others issues, users are always encouraged and recommended to consult their health professional before acting on any information seen on site.

**9. BIBLIOGRAPHY**

**9.1 REFERENCES**

**9.2 LIST OF WEBSITES:**