



American International University-Bangladesh (AIUB)

Department of Computer Science
Faculty of Science & Technology (FST)
Fall 19_20

Section: J

Group No: 01

Doctor Appointment & Medicine

A software Engineering project submitted
By

Arshad Hassan Aywon [18-36601-1] (@Aywon)

Bushra Amana (18-36568-1)(BushraAmana)

Aseya Khanom (17-35394-3) (aseyaava)

Shafin Talukder (18-36588-1) (nill1279)

The project will be evaluated for the following Course Outcomes

CO3: Choose appropriate software engineering model in a software development environment	Total Marks
Project Background Analysis (needs, goal, benefits, etc.) [5Marks]	
Appropriate Process Model Selection [5Marks]	
Argumentation for model selection with Evidence [5Marks]	
Completeness, Spelling, Grammar and Organization of the Answer [5Marks]	
CO4: Explain the roles and their responsibilities in the software project management activities	Total Marks
Content Knowledge (e.g. System Requirements, System Design) [5Marks]	
Project Role identification [5Marks]	
Responsibility Description [5Marks]	
Completeness, Spelling, grammar and Organization of the Answer [5Marks]	

Contents

1. Problem Domain.....	Error! Bookmark not defined.
1.1 Background to the Problem.....	Error! Bookmark not defined.
1.2 Solution to the Problem	Error! Bookmark not defined.
1.3 Existing / Related Solutions	Error! Bookmark not defined.
2. Software development lifecycle.....	Error! Bookmark not defined.
2.1 Process Model.....	4
2.1.1 Agile.....	4
2.1.2 Scrum.....	5
2.1.3 FDD	5
2.2 Your Process model.....	6
2.3 Time box concept in Agile/Sprint in scrum.....	8
3. PRODUCT AND PROJECT DESCRIPTION.....	8
3.1 Stakeholders	8
3.2 System Features.....	8
3.3 System Quality Attributes.....	10
3.4 System Architecture.....	11
3.5 System Interface	16
4. Project Management	22
4.1 Project Roll Identification and Responsibilities	22
4.2 COCOMO model.....	23
4.3 Earned value analysis (eva).....	23
5. References.....	23

1. PROBLEM DOMAIN

1.1 Background to the Problem

The application is about shipping the rare medicines from abroad and to the people in our country. There are medicines which can only be found in a few fixed countries. The current scenario of our country is if one wants a particular medicine from abroad, he must have to go and get the medicine for a limited time period all by himself. By thus a huge amount of not only money but also time and energy is wasted.

The second thing our application will do is it will allow you to make an appointment with doctors. Now this is not a new idea at all in our country and there are a lot of websites for appointment as well. But the problem is one website allows under only one particular medical doctor to make the appointment.

1.2 Solution to the Problem

The problem of the crisis of rare unavailable medicine from abroad can be solved using a simple shipping system to buy medicines from abroad. So, we are thinking about making an application system in which we will make the rare and expensive medicines available anytime from all over the world. As we want to solve this problem, we need to create a system which will save our time & money and provide medicine as soon as possible. The goal of our application is to make a way that can provide rare medicine available anytime.

For the second problem of making an appointment the concept of our application which will bring the change is one can make the appointment with any doctors who is appointed in any medical contra's in all over the country. So, the theme is we are getting all the doctors list in just one application. By this the appointment system will be much more efficient to use.

So the benefits of the application is

- Save a huge amount of time, energy and money.
- Make the wanted unavailable medicines available anytime.
- Get the appointment sitting anywhere in the country.

1.3 Existing / Related Solutions

The existing solution of shipping the rare medicines is currently not available in our country at all. This is why we are proposing the shipping proposal. And also, there is no existing solution for the concept of making the appointment in our country.

2. SOFTWARE DEVELOPMENT LIFE CYCLE

2.1 Process Model

2.1.1 Agile

“Plan-driven methods work best when developers can determine the requirements in advance . . . and when the requirements remain relatively stable, with change rates on the order of one percent per month.”~ *Barry Boehm*

- ☐ Agility is the ability to create and respond to change in order to profit in a turbulent business environment
- ☐ Subset of iterative and evolutionary methods

Iterative Products

- Each iteration is a self-contained, mini-project with activities that span requirements analysis, design, implementation, and test
- Leads to an iteration release (which may be only an internal release) that integrates all software across the team and is a growing and evolving subset of the final system
- The purpose of having short iterations is so that feedback from iterations N and earlier, and any other new information, can lead to refinement and requirements adaptation for iteration N + 1

Time Box

- ☐ The pre-determined iteration length serves as a timebox for the team.
- ☐ Scope (set of tasks) is chosen for each iteration to fill the iteration length.
- ☐ Rather than increase the iteration length to fit the chosen scope, the scope is reduced to fit the iteration length.

Agile method info:

- ☐ Agile methods are considered
 - Lightweight (do not concentrate on the whole s/w development at once)
 - People-based rather than Plan-based
- ☐ Several agile methods
 - No single agile method
 - Different agile methods can be combined in s/w development (Hybrid)
- ☐ No single definition. Agile Manifesto closest to a definition
 - Set of principles
 - Developed by Agile Alliance

Agile Assumption:

- ☐ It is difficult to predict in advance which software
 - requirements will persist and which will change
 - It is equally difficult to predict how customer priorities will change as the project proceeds
- ☐ Design and construction are interleaved in many types of software. That is, both activities should be performed tightly so that design models are proven as they are created. It is difficult to predict how much design is necessary before construction is used to prove the design.
- ☐ Analysis, design, construction, and testing are not as predictable (from a planning point of view) as we might like.

2.1.2 Scrum

SCRUM process includes three phases

- Pre-game
- Development (game phase)
- Post-game

1. Pre-game:

- Definition of the system being developed
- A Product Backlog list is created containing all the requirements that are currently known
- The requirements are prioritized and the effort needed for their implementation is estimated
- The product Backlog list is constantly updated with new and more detailed items, as well as with more accurate estimations and new priority orders
- Planning also includes the definition of the project team, tools and other resources, risk assessment and controlling issues, training needs and verification management approval

2. Development phase:

- ☐ This phase is treated as a "black box" where the unpredictable is expected
- ☐ The system is developed in Sprints
 - Sprints are iterative cycles where the functionality is developed or enhanced to produce new increments.
 - Each Sprint includes the traditional phases of software development: requirements, analysis, design, evolution and delivery phases.
 - One Sprint is planned to last from one week to one month.

3. Post-game:

- ☐ This phase is entered when an **agreement** has been made such as the requirements are completed.
- ☐ In this case, **no more items and issues** can be found nor can any new ones be invented.
- ☐ The system is now ready for the **release** and the preparation for this is done during the post-game phase, including the tasks such as the integration, system testing and documentation.

2.1.3 FDD

Feature Driven Development (FDD)

- FDD is an agile software development process
- FDD uses a short-iteration model
- FDD was created to easily scale to much larger projects and teams

Info:

- FDD delivers the system feature by feature
- Features are to be “small” in the sense they will take no more than two weeks to complete Features that appear to take longer are to be broken up into a set of smaller features. Two weeks is the maximum, most features take less time (1 - 5 days)
- Feature naming template:

<action> the <result> <by|for|of|to> a(n) <object>

- Examples: Calculate the total of a sale
Validate the password of a user

Process:

- ☐ Process #1: Develop an Overall Model
- ☐ Process #2: Build a Features List
- ☐ Process #3: Plan By Feature
- ☐ Process #4: Design By Feature
- ☐ Process #5: Build By Feature

Process #1: Develop an Overall Model

- ☐ Form a modeling team
- ☐ Domain walk-through
- ☐ Build High-level object model
- ☐ Record Notes

Goal - for team members to gain a good, shared understanding of the problem domain and build a foundation

Process #2: Build a Features List

- All Features are organized in a three level hierarchy

Process #3: Plan by Feature

- Construct initial schedule
 - ☐ Formed on level of individual features
 - ☐ Prioritize by business value
 - ☐ Also consider dependencies, difficulty, and risks
- Assign responsibilities to team members
 - ☐ Determine Class Owners
 - ☐ Assign feature sets to chief programmers

Process #4: Design by Feature

- Form Feature Teams
- Team members collaborate on the full low level analysis and design
- Certain features may require teams to bring in domain experts
- Teams need to update the model artifact to support their changes

Process #5: Build by Feature

- Implement designed feature
- Test feature
 - >Unit-level >Feature-level
- Mandated Code Inspections (formal review with checklist)
- Integrate with regular build

2.2 Our Process Model

To analysis the development of the software we want to go with **scrum development method**.

SCRUM process includes three phases and they are Pre-game, Development (game phase) and Post-game.

The pre-game phase includes two sub-phases:

Planning:

- A Product Backlog list is created containing all the requirements that are currently known.
- The requirements are prioritized and the effort needed for their implementation is estimated.

- The product Backlog list is constantly updated with new and more detailed items, as well as with more accurate estimations and new priority orders.

Architecture:

- The high level design of the system including the architecture is planned based on the current items in the Product Backlog
- In case of an enhancement to an existing system, the changes needed for implementing the Backlog items are identified along with the problems they may cause
- A design review meeting is held to go over the proposals for the implementation and decisions are made on the basis of this review.

In the Development phase, this phase is treated as a "black box" where the unpredictable is expected. Unpredictable means any kind of requirements in all of a sudden. The system is developed in Sprints. Sprints are iterative cycles where the functionality is developed or enhanced to produce new increments.

The post game phase is entered when an **agreement** has been made such as the requirements are completed. In this case, **no more items and issues** can be found nor can any new ones be invented. The system is now ready for the **release** and the preparation for this is done during the post-game phase, including the tasks such as the integration, system testing and documentation.

We are using scrum as our software development model because it has lots of advantages. The sprint process allows for "good enough" development that results in a saleable product even while the project is in full swing. This incremental delivery system shortens the time to market and may result in higher revenue, as each completed backlog represents a new release of the product. In addition, reviewing each sprint before moving to the next means that testing is conducted throughout the process, which allows teams to change the scope or direction of the project at any point. Although the deadline and budget are fixed variables, the project requirements are not. In fact, stakeholders and participants anticipate changes along the way. The product owner's involvement in the project management facilitates these changes. There are other models like waterfall, v-model, incremental, FDD, Xp etc. We are not using waterfall or V-model because our requirements are not clear and there will be changes in future. But in scrum requirements are not fixed as we can change it. Also, we are not using FDD or Xp though scrum, FDD, XP are agile methodologies but they have some differences. Both approaches provide agility but in different ways. We find when practicing Scrum, we may perform some up-front design before we delve into code which is similar to what's advocated by FDD. However, we find feedback loops in Scrum are much quicker than in FDD. The loop between design and development phases can be much shorter and it's easier to change tack if we find the direction, we are following is incorrect. Also, it has more formal requirements and steps than Scrum from a development perspective. As our project is for business purposes therefore, we think scrum model will be suitable for our project.

2.3 Time box concept in Agile/Sprint in scrum

Time Box

- ☐ The pre-determined iteration length serves as a timebox for the team.
- ☐ Scope (set of tasks) is chosen for each iteration to fill the iteration length.
- ☐ Rather than increase the iteration length to fit the chosen scope, the scope is reduced to fit the iteration length.

3. PRODUCT AND PROJECT DESCRIPTION

3.1 Stakeholders

Stakeholders:

Stakeholders are those with any interest in the project's outcome. They are typically the members of a project team, project managers, executives, project sponsors, customers, and users. Stakeholders are people who are invested in the project and who will be affected by the project at any point along the way, and their input can directly impact the outcome.

Internal Stakeholders are groups or individuals who work within an organization or project. So, for our project **the internal stakeholders are admin, patient and salesman.**

External stakeholders are those who are not directly linked to the projects but can influence the activities of the project through various ways. So, for our project **the internal stakeholders are Hospitals, Medicine companies and government.**

3.2 System Features

- **Registration:**
 1. The software will allow the user to create an account through some personal information's by which they can later log in to the software.
 2. One number and g-mail can be used for only one account registration. The system will not allow to use the same number or g-mail for registering various accounts.
- **Log in:**

1. The software will allow the user to log in by using the information's that have been provided during registration.
2. Wrong info will not allow to in.
3. So far there is still no restriction that if one provides wrong info more than 2 or 3 times the user account will be blocked.

- ***Appointment:***

1. This portion will allow the user to make the appointment with the doctor.

- ***Doctors Information:***

1. By entering into this feature the user can choose under whom he/she want to make the appointment.
2. The user can also make the appointment by searching a specific hospital name.

- ***Confirm appointment:***

1. The user can confirm the appointment by selecting the appointment time.

- ***Search Medicine:***

1. This feature will allow the user to search the medicine that the user wants to ship.

- ***Medicine Status:***

1. This feature will give a feedback to the user that weather the ordered medicine is available or not.

- ***Available:***

1. If the medicine is available then the feature will request the user to provide the quantity.
2. Base on the quantity the user will then show the price and delivery date.
3. Then it will allow the user to confirm the shipping.

- ***Unavailable:***

1. If the medicine is unavailable then the feature will request the user if the user still want to order the medicine.
2. After confirming, the feature will request to provide the quantity.
3. Base on the quantity the user will then show the price and delivery date.
4. Then it will allow the user to confirm the shipping.

- ***Payment Method:***

1. This feature will provide various ways of payment to the user. The user can pay through card, bkaash, nagad, rocket or make the payment on cash on delivery.
2. The feature will be exited after confirming.

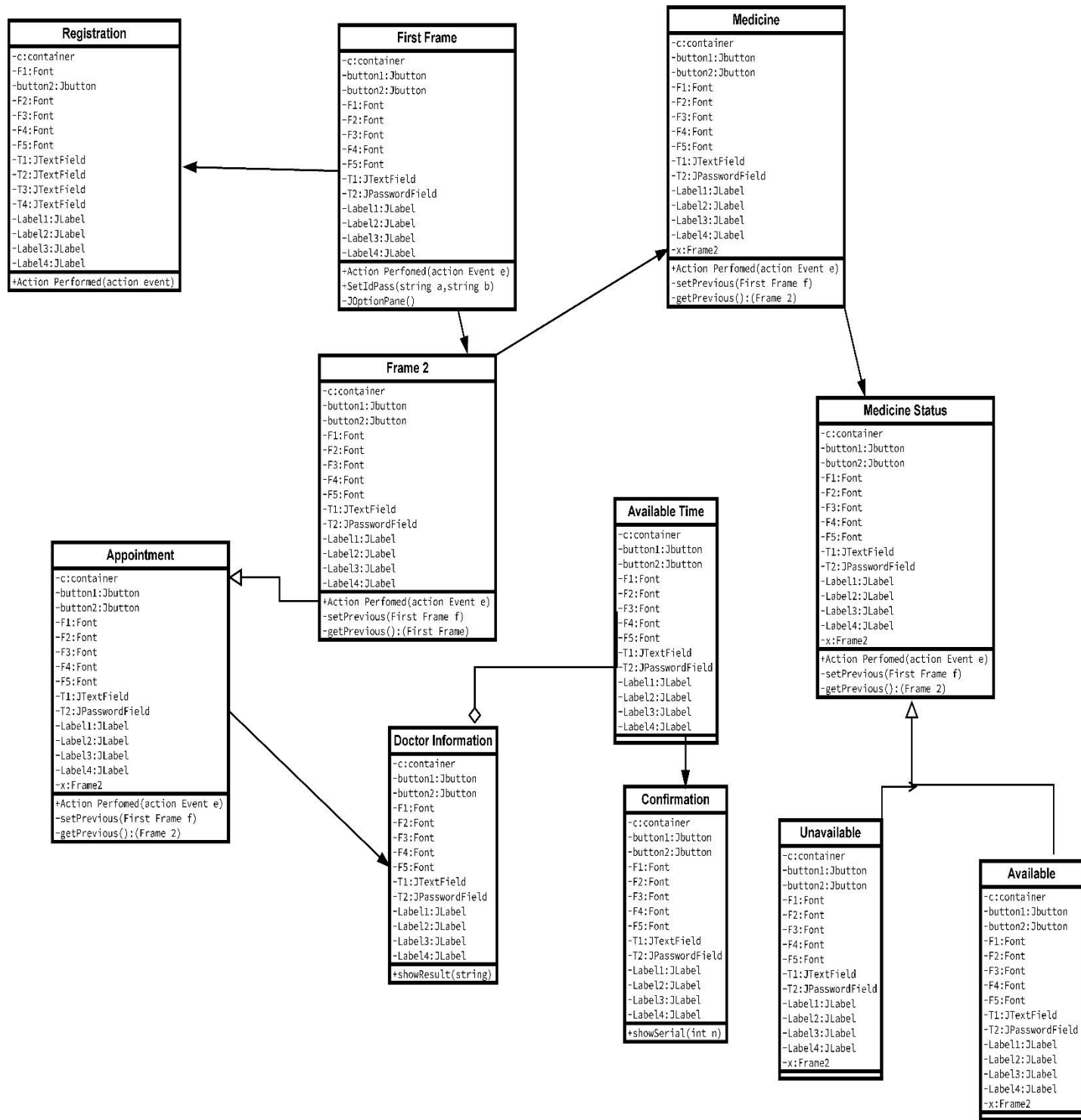
- ***Give Address:***

1. This feature is for providing the address where the product will be shipped.
2. After confirming, the feature will then confirm the order.

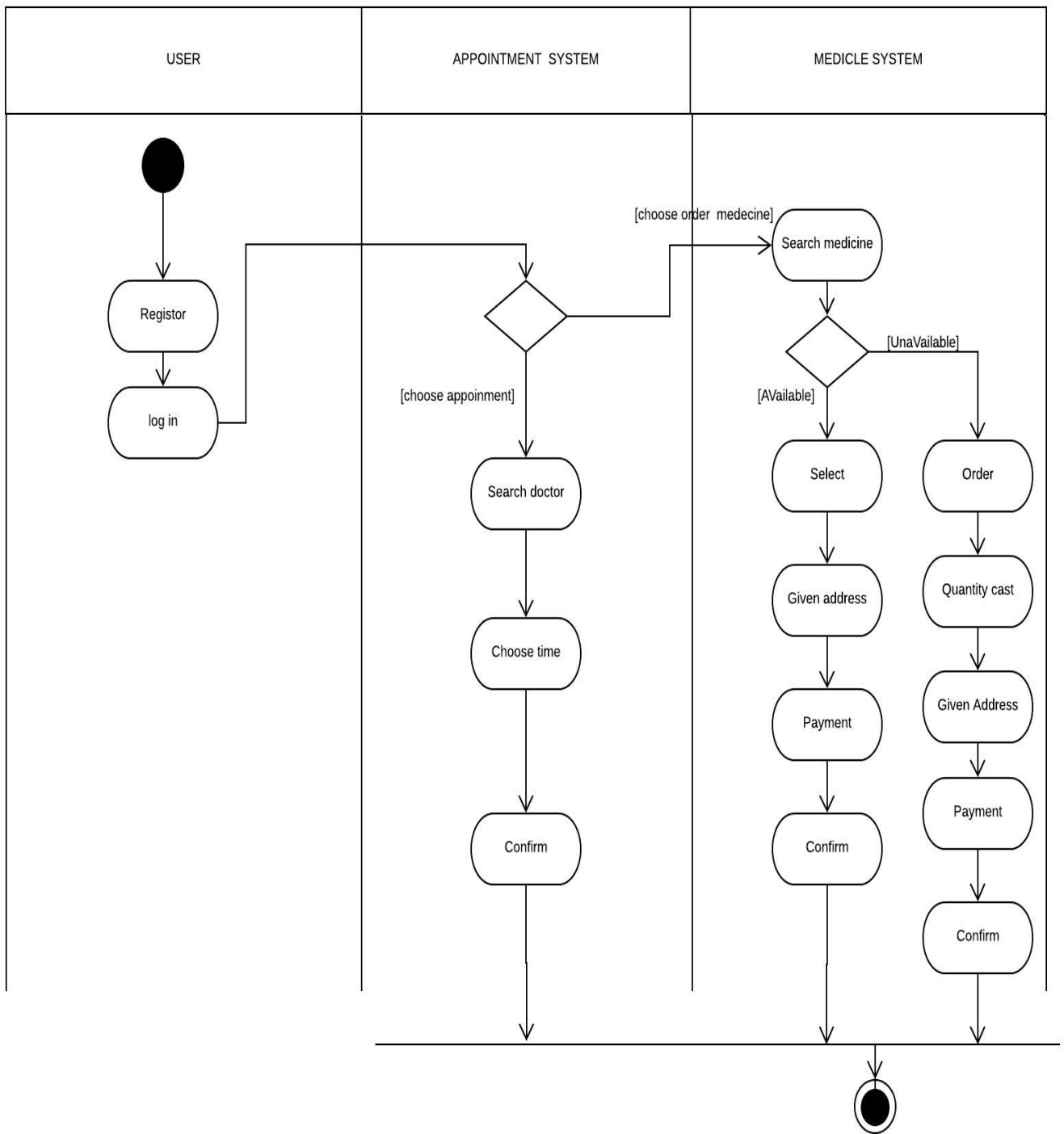
3.3 System Quality Attributes

- **Usability:** A trained user shall be able to submit a complete request for an appointment selected from a vendor catalog and also can order medicine in an average of four and a maximum of six minutes.
- **Flexibility:** A maintenance programmer who has at least six months of experience supporting this application shall be able to make a new copy output available to the feature, including code modifications and testing, with no more than one hour.
- **Robustness:** If the editor fails before the user saves the file, the editor shall be able to recover all changes made in the file being edited up to one minute prior to the failure the next time the same user starts the program.
- **Performance:** Every Web page shall download in 15 seconds or less over a 50 KBps modem connection.

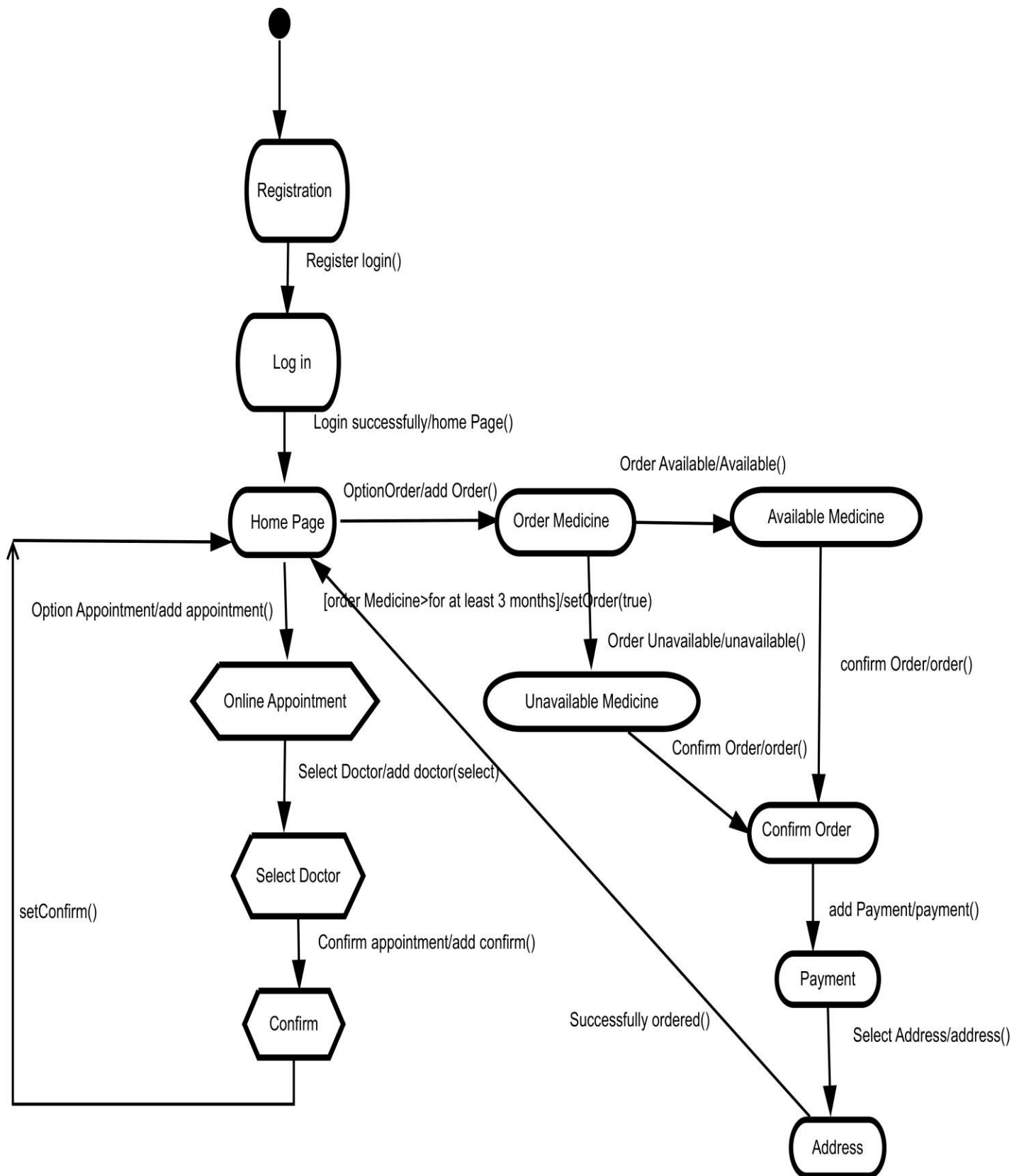
3.4 System Architecture



Class Diagram

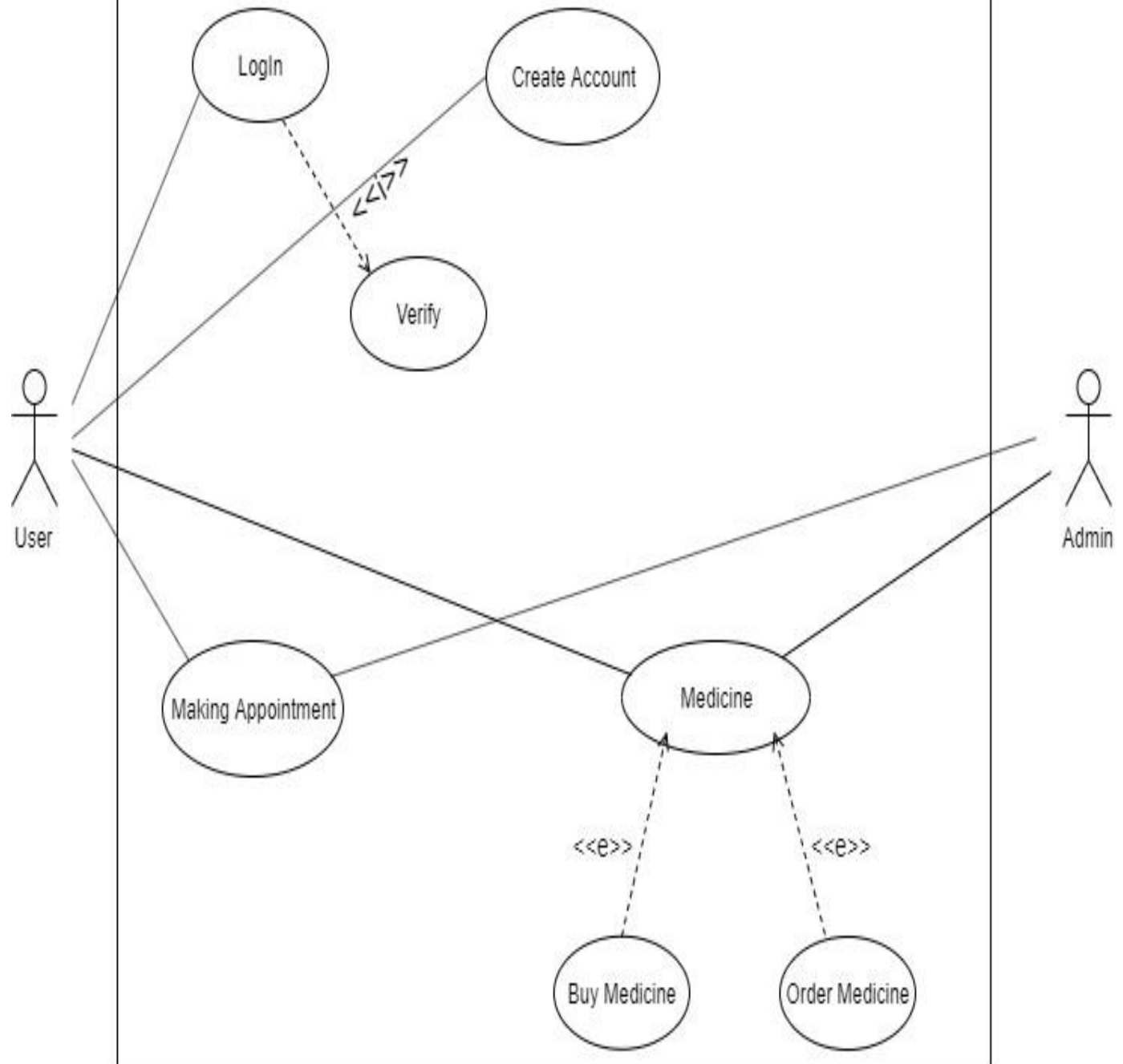


Activity Diagram

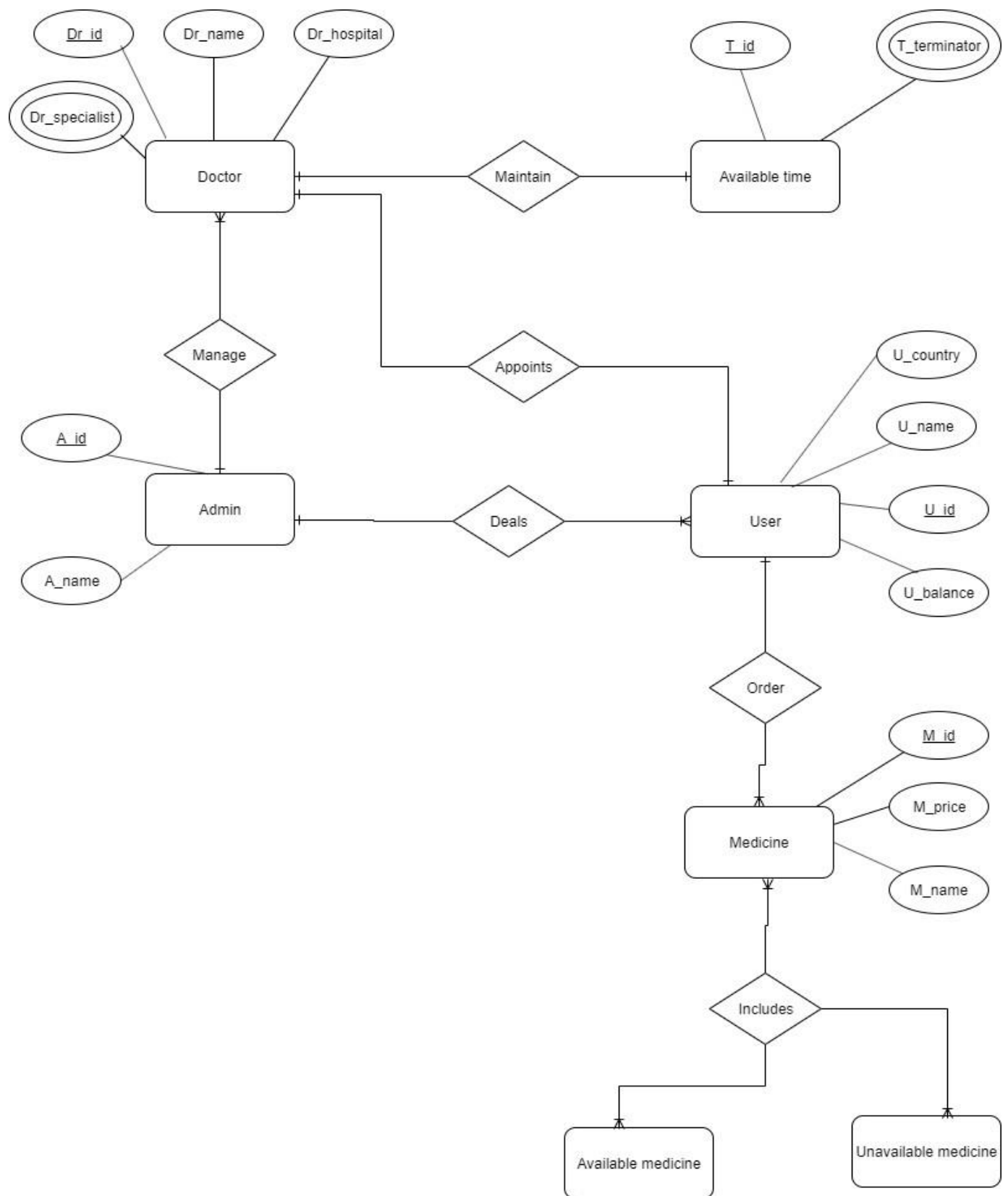


State Chart Diagram

DOCTOR APPOINTMENT & MEDICINE



Use Case Diagram



ER Diagram

3.5 System Interface

Welcome to **Doctor Appointment & Medicine**

Phone Number:

Password:

[Already Have an Account?](#)

Personal Information

Name:

Phone Number:

Gmail:

Set Password:

Doctor Appointment & Medicine

Appointment

Order Medicine

Back

Appointment

Specialist:

Doctor's Name:

Search

Previous

Doctor's Information

Doctor's Name:

Hospital:

Visiting Hours:

See Available Time

Previous

Select Your Time

Select Time



Confirm Appointment

Previous

Search Medicine

Medicine:

Search

Previous

Medicine Status

Available

Unavailable

Previous

Available

Medicine Name:

Quantity:

Price:

Delivery Date:

Previous

Confirm Order

Unavailable

Medicine Name:

Quantity:

Price:

Delivery Date:

Previous

Order

Select Payment Method

Select Payment Method

Select Payment Method

Card

bKash

Nagad

Rocket

Cash On Delivery

Confirm Payment

Previous

Give Address

Previous

Confirm Address

4 Project Management

RISK: Risk of our project is whether it can meet customers satisfaction and fulfill the requirements. Also cost is the another risk. We can be sure that our project can be done by the budget. Another thing is uncertain that our project will be easy to change, adapt and enhance. Last one is schedule risk whether we can finish it within our deadline.

Schedule: We have 18 tasks, but we completed 14 tasks. We are total 4 person work each day to complete.

4.1 Project Roll Identification and Responsibilities

All the roles in the project management activities in software development are define below:

→ Scrum Master

- ✦ Scrum Master is responsible for ensuring that the project is carried through according to the practices, values, and rules of Scrum and that it progresses as planned.
- ✦ Scrum Master interacts with the project team as well as with the customer and the management during the project.

→ Product Owner

- ✦ Product Owner is officially responsible for the project, managing, controlling, and making visible the Product Backlog list.
- ✦ He is selected by the Scrum Master, the customer, and the management.
- ✦ He makes the final decisions of the tasks related to product Backlog.

→ Scrum Team

- Scrum Team is the project team that has the authority to decide on the necessary actions and to organize itself in order to achieve the goals of each Sprint.
- The scrum team is involved, for example, in effort estimation, creating the Sprint Backlog, reviewing the product Backlog list and suggesting impediments that need to be removed from the project.

→ Customer

- Customer participates in the tasks related to product Backlog items for the system being developed or enhanced.

→ Management

- Management is in charge of final decision making, along with the agreements, standards, and conventions to be followed in the project.
- Management also participates in the setting of goals and requirements.

4.2 COCOMO model

EFFORT:

$$PM = 2.4 * (3000/1000)^{1.05} = 7.61$$

$$DM = 2.50 * (7.61)^{0.38} = 5.5058$$

$$\text{Required number of people} = (7.61/5.40) = 1.90 \text{ (almost 2)}$$

4.3 Earned value analysis(EVA)

The project has 12 planned work tasks that are estimated to require 40 person days to complete. 4 task have been completed. However our project schedule indicates that 5 tasks should be completes. The following schedule data (in person days) are available.

Task	Planned effort		Actual effort	
1	8.0		9.5	
2	12.0	BCWP=29.5	11.0	ACWP=45.5
3	5.0		10.5	
4	4.5		14.5	
5	10.0			

$$BAC = 40$$

$$SPI = BCWP/BCWS = (29.5/39.5) = 0.7468$$

$$SV = BCWP-BCWS = (29.5-39.5) = -10$$

$$CPI = BCWP/ACWP = (29.5/45.5) = 0.64835$$

$$CV = BCWP-ACWP = (29.5-45.5) = -16$$

$$\% \text{ Schedule for completion} = BCWS/BAC = (39.5/40) = 0.98$$

$$\% \text{ Complete} = BCWP/BAC = (29.5/40) = 0.74$$

5 References

[1] Roger Pressman. 2009. Software Engineering: A Practitioner's Approach (7th. ed.). McGraw-Hill, Inc., USA.

[2] SLIDES