DeepCells – CANCER PREDICTION SYSTEM

RUHUL QUDDUS TAMIM

UNIVERSITI TEKNOLOGI MALAYSIA

UNIVERSITI TEKNOLOGI MALAYSIA

DECLARATION OF THESIS / UNDERGRADUATE PROJECT REPORT AND COPYRIGHT

Author's full name : Ruhul Quddus Tamim

Date of Birth : 17 April 1999

Title : DeepCells – Cancer Prediction System

Academic Session : 2021/2022 - 2

I declare that this thesis is classified as:

☐	**CONFIDENTIAL**	(Contains confidential information under the Official Secret Act 1972)*
☐	**RESTRICTED**	(Contains restricted information as specified by the organization where research was done)*
☒	**OPEN ACCESS**	I agree that my thesis to be published as online open access (full text)

1 I acknowledged that Universiti Teknologi Malaysia reserves the right as follows:

2 The thesis is the property of Universiti Teknologi Malaysia

3 The Library of Universiti Teknologi Malaysia has the right to make copies for the purpose of research only.

4 The Library has the right to make copies of the thesis for academic exchange.

Certified by:

_____	_____
SIGNATURE OF	**SIGNATURE OF**

NOTES : If the thesis is CONFIDENTIAL or RESTRICTED, please attach with the letter from the organization with period and reasons for confidentiality or restriction

"I hereby declare that we have read this thesis and in my opinion this thesis is suffcient in term of scope and quality for the award of the degree of Bachelor of Computer Science (Software Engineering)"

Signature	:	_____
Name of Supervisor	:	Dr Halinawati Binti Hirol
Date	:	9 PM 6 JULY 2022

DeepCells – CANCER PREDICTION SYSTEM

RUHUL QUDDUS TAMIM

A thesis submitted in fulfilment of the requirements for the award of the degree of

Bachelor of Computer Science (Software Engineering)

School of Computing

Faculty of Engineering

Universiti Teknologi Malaysia

JULY 2022

DECLARATION

I declare that this project report entitled “DeepCells – Cancer Prediction System ” is the result of my own research except as cited in the references. The project report has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

Signature	:	tamim
Name	:	RUHUL QUDDUS TAMIM
Date	:	9 PM 6 JULY 2022

DEDICATION

This project is dedicated to my father, who taught me to be honest at any cost. My father also taught me discipline and hard work is the key to success. From my beginning to this stage of my life he has been with me as roof to protect me from external things. It is also dedicated to my mother, who taught me the kindness. The soft warm heart of my mother showed me the kindness and softness can achieve so many powerful things. I am proud that I was born to them, and I am proud of my upbringing. I dedicate this project to them; they are my world. May God always keep blessing them forever and ever.

ACKNOWLEDGEMENT

In preparing this project, I was in contact with many people, researchers, academicians, and practitioners. They have contributed towards my understanding and thoughts. I wish to express my sincere appreciation to my main project supervisor, Dr Halinawati Binti Hirol, for encouragement, guidance, critics and friendship. Without her continued support and interest, this project would not have been the same as presented here. Since the library is also closed during the Covid-19 pandemic, I am also thankful to Universiti Teknologi Malaysia (UTM) to maintain the digital facilities such as university google email, and proxy service to online databases goes up without experiencing the service goes down.

My fellow undergrads student should also be recognised for their support. Especially Md Yusuf Bin Forkan. He was one of my biggest helper and supporter. I cannot thank him enough. May God bless you. My sincere appreciation also extends to all my classmates and others who have aided at various occasions. Their views and tips are useful indeed. Unfortunately, it is not possible to list all of them in this limited space. I am grateful to all my family member.

ABSTRACT

A major challenge facing healthcare organizations (hospitals, medical centers) is the provision of quality services at affordable costs. Quality service implies diagnosing patients correctly and administering treatments that are effective. Poor clinical decisions can lead to disastrous consequences which are therefore unacceptable. Hospitals must also minimize the cost of clinical tests. They can achieve these results by employing appropriate computer-based information and/or decision support systems. Most hospitals today employ some sort of hospital information systems to manage their healthcare or patient data. These systems typically generate huge amounts of data which take the form of numbers, text, charts and images. Unfortunately, these data are rarely used to support clinical decision making. There is a wealth of hidden information in these data that is largely untapped. This raises an important question: “How can we turn data into useful information that can enable healthcare practitioners to make intelligent clinical decisions?” Although data mining has been around for more than two decades, its potential is only being realized now. Data mining combines statistical analysis, machine learning and database technology to extract hidden patterns and relationships from large databases. The two most common modeling objectives are classification and prediction. Classification models predict categorical labels (discrete, unordered) while prediction models predict continuous-valued functions. Decision Trees and Neural Networks use classification algorithms while Regression, Association Rules and Clustering use prediction algorithms.

Naive Bayes or Bayes‟ Rule is the basis for many machine-learning and data mining methods. The rule (algorithm) is used to create models with predictive capabilities. It provides new ways of exploring and understanding data. It learns from the “evidence” by calculating the correlation between the target (i.e., dependent) and other (i.e., independent) variables. In Weighted Associative Classifier (WAC), different weights are assigned to different attributes according to their predicting capability. Weighted Associative Classifier (WAC) is a new concept that uses Weighted Association Rule for classification. Weighted ARM uses Weighted Support and Confidence Framework to extract Association rule from data repository. The WAC has been proposed as a new Technique to get the significant rule instead of flooded with insignificant relation.

ABSTRAK

Cabaran utama yang dihadapi oleh organisasi penjagaan kesihatan (hospital, pusat perubatan) ialah penyediaan perkhidmatan berkualiti pada kos yang berpatutan. Perkhidmatan berkualiti membayangkan mendiagnosis pesakit dengan betul dan mentadbir rawatan yang berkesan. Keputusan klinikal yang buruk boleh membawa kepada akibat yang buruk yang oleh itu tidak boleh diterima. Hospital juga mesti meminimumkan kos ujian klinikal. Mereka boleh mencapai keputusan ini dengan menggunakan maklumat berasaskan komputer dan/atau sistem sokongan keputusan yang sesuai. Kebanyakan hospital hari ini menggunakan beberapa jenis sistem maklumat hospital untuk mengurus data penjagaan kesihatan atau pesakit mereka. Sistem ini biasanya menjana sejumlah besar data yang berbentuk nombor, teks, carta dan imej. Malangnya, data ini jarang digunakan untuk menyokong membuat keputusan klinikal. Terdapat banyak maklumat tersembunyi dalam data ini yang sebahagian besarnya belum diterokai. Ini menimbulkan persoalan penting: "Bagaimanakah kita boleh menukar data menjadi maklumat berguna yang membolehkan pengamal penjagaan kesihatan membuat keputusan klinikal yang bijak?" Walaupun perlombongan data telah wujud selama lebih daripada dua dekad, potensinya hanya direalisasikan sekarang. Perlombongan data menggabungkan analisis statistik, pembelajaran mesin dan teknologi pangkalan data untuk mengekstrak corak dan perhubungan tersembunyi daripada pangkalan data yang besar. Dua objektif pemodelan yang paling biasa ialah pengelasan dan ramalan. Model pengelasan meramalkan label kategori (diskrit, tidak tertib) manakala model ramalan meramalkan fungsi bernilai berterusan. Pokok Keputusan dan Rangkaian Neural menggunakan algoritma pengelasan manakala Regresi, Peraturan Persatuan dan Pengelompokan menggunakan algoritma ramalan.

Naive Bayes atau Bayes‟ Rule adalah asas bagi banyak kaedah pembelajaran mesin dan perlombongan data. Peraturan (algoritma) digunakan untuk mencipta model dengan keupayaan ramalan. Ia menyediakan cara baharu untuk meneroka dan memahami data. Ia belajar daripada "bukti" dengan mengira korelasi antara sasaran (iaitu, bergantung) dan pembolehubah lain (iaitu, bebas). Dalam Weighted Associative Classifier (WAC), pemberat yang berbeza diberikan kepada atribut yang berbeza mengikut keupayaan meramalnya. Weighted Associative Classifier (WAC) ialah konsep baharu yang menggunakan Weighted Association Rule untuk pengelasan. Weighted ARM menggunakan Weighted Support and Confidence Framework untuk mengekstrak peraturan Persatuan daripada repositori data. WAC telah dicadangkan sebagai Teknik baharu untuk mendapatkan peraturan penting dan bukannya dibanjiri dengan hubungan yang tidak penting.

TABLE OF CONTENTS

LIST OF FIGURES

FIGURE NO. TITLE PAGE

LIST OF ABBREVIATIONS

CPS	-	Cancer Prediction System
UI	-	User Interface
DB	-	Database
RAAD	-	Rapid Architected Application Development
IT	-	Information Technology
MVC	-	Model View Controller
BVA	-	Boundary Value Analysis
EP	-	Equivalence Partitioning
SQL	-	Structured Query Language
WHO	-	World Health Organization
ML	-	Machine Learning
SRS	-	Software Requirements Specification
SDD	-	Software Design Documentation
STD	-	Software Testing Documents
KNN	-	K-Nearest Neighbours
NN	-	Neural Network
IDE	-	Integrated Development Environment
FYP	-	Final Year Project

1

INTRODUCTION

1.1 Introduction

Using supervised learning techniques, classification determines the intended class for each instance of data. As part of data mining, classification is the process of assigning things to certain classifications. For classification, the primary goal is to reliably forecast the target class for each state in the data By using classification, a data analyst may find previously unknown data models. It's a crucial aspect that helps us choose and decide between many organizations. Medical diagnosis, picture and pattern recognition, problem detection in business, and a slew of other uses fall under the classification category. The effectiveness of a classification model is determined by the model's interpret-ability and accuracy [1]. Cancer is the leading cause of mortality in the world, accounting for more than half of all deaths. It is a disease that may affect any area of the body and can spread to several other sections due to the abnormal proliferation of cells in any of the tissues or organs. It is not necessary for all tumour to be malignant, thus some do not spread throughout the body. A tumour, irregular bleeding, a longer cough, and increased weight loss are all signs of cancer. Early identification of cancer is the most effective means of reducing cancer-related fatalities. A reliable and precise diagnostic approach is needed to differentiate between benign and malignant tumours without the need for a surgical sample in the early stages of diagnosis.

1.2 Problem Background

According to the World Health Organization (WHO), cancer will cost the lives of 9.6 million people worldwide in 2022. (World Health Organization). In addition, there are around 0.3 million new cancer cases diagnosed in children between the ages of zero and 19. One of the most lethal illnesses a person may develop is cancer. At least half (50 percent) of all cancers may be avoided and cured if they are discovered early enough. Otherwise, a dangerous scenario might arise, perhaps resulting in death.

Every year, tens of thousands of individuals die from lung cancer, which is a prevalent hereditary illness. It is caused by a variety of factors, including smoking cigarettes, which chemically activate the oncogenes and deactivate the tumor suppressor in the normal lung cell and generate mutations that result in tumors in the lungs. This necessitates the use of reliable diagnostic techniques that are both quick and sensitive. The need to create a system or technology that can assist physicians diagnose cancer at an early stage where it may be properly treated has become very critical.

An intelligent system is essential for solving this challenge using artificial intelligence. Using the Naive Bayes Machine Learning method, a Cancer Prediction System has been proposed. Based on previous data sets, this system use probabilistic and optimization approaches to bring forth a conclusion. Doctors and pathologists can save many lives by using this assessment method to discover cancer at an early stage when it may be treated and cured.

1.3 Project Aim

The aim of this project is to classify the different types of cancer data as accurately as possible by almost (98 – 99)% in order to predict the symptoms for cancer types.

1.4 Project Objectives

I. To collect the data-sets for types of cancer provided by organizations, e.g. UNICEF, World Health Organization(WHO).

II. To pre-process the data using data cleaning tools and prepare the data for training data-set.

III. To predict/test the data on the trained result and classify the data using Naive Bayes Machine Learning algorithm and display the class label.

1.5 Project Scope

The software product is DeepCells – Cancer Detection System. Generally, this system will allow the users to input necessary details and see the type of cancer predicted at an early stage.

What the system can do:

I Allows the admin to login and view or add training data for the algorithm to analyze and predict results.

II Allows the admin to add or view specialist details with respect to types of cancer.

III Allows the admin to view all the registered users' details and their feedback.

IV Allows the user to register with their basic details and predict cancer by inputting all the necessary details and the system will accordingly view doctor to consult.

V Allow the user to view various doctor based on the predicted cancer.

What the system cannot do:

I. The system cannot let the patient view the training data.
II. The system cannot let the patient view other patient's profile.

The objective and goals of this system are to upgrade the current problem scenario in the system and design the system which meets the administrative, organizational, and management requirements of hospital. In addition, CPS also allows the users to access the system anytime, anywhere with the internet connection

1.6 Project Importance

Cancer research has seen a steady change throughout the last several decades. Scientists used a variety of strategies, including early screening, to identify cancers before they had a chance to spread. In addition, they have devised novel methods for predicting cancer therapy outcomes at an early stage. With the development of new technology in the realm of medicine, enormous volumes of cancer data have been gathered and are accessible to the medical research community. However, the precise prognosis of a disease outcome is one of the most exciting and demanding jobs for clinicians. As a consequence, ML approaches have become a popular tool for medical researchers. The techniques applied in this project may find and identify patterns and links between them, from complicated information, while they are able to successfully forecast future outcomes of a cancer kind.

1.7 Report Organization

The first chapter describes the problem statement and the proposed solution of the cancer prediction system. It also discusses the background study and the how the problem is identified as well as also the impact of the system or benefits in future use by hospitals.

The literature review covers the basic theories regarding the project and previous system that has been developed by other developers. It has also explained why this system is much more preferable than the other existing system.

The system methodology and requirements analysis describes the project will be done within the deadline. The way of approaching the project, project estimation, type of technology used for developing the project and making the system architecture, designing the database according to the system methodology.

Lastly, after developing the system, the system will be tested in order to assure the quality of the project. Various testing will be done from Black-box testing to White-Box testing until User Acceptance testing. After testing, then finally the system can be deployed and available to everyone.

2

LITERATURE REVIEW

2.1 Introduction

This chapter was written primarily for the aim of better understanding the study's goals and objectives, as well as the significance of the work in light of other research on cancer prediction systems. The literature review's findings will also be discussed in detail, including current research that examine the components of cancer prediction in the context of the actual world. A few examples from the literature, as well as related studies, will be cited to support the findings of this research on system development. A map of the subjects addressed in the literature review will help students better comprehend the scope of the project.

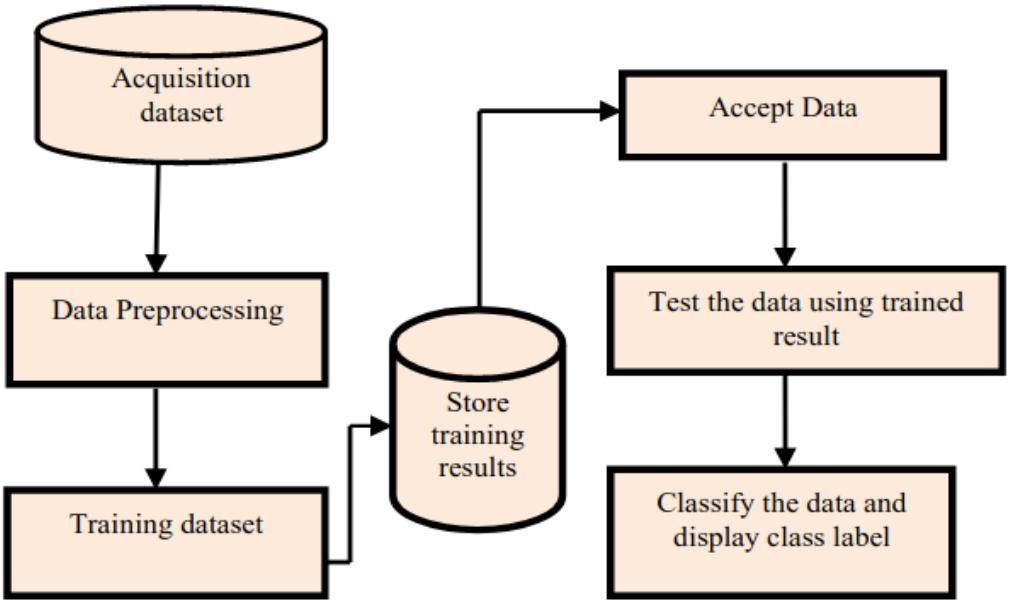


Figure 1 Proposed System

2.2 Naive Bayesian Classifier

The Nave Bayes Classifier is one of the most straightforward, efficient, and quick Classification algorithms in machine learning. It makes predictions based on the probability that an item will be present or absent since it is a probabilistic classifier.

2.2.1 Calculate Probability

The Bayes Theorem is the foundation of the Naive Bayesian Classifier. The naive Bayesian model assumes that all variables are mutually independent. A tuple and its associated class label may be defined as D, the training data set. N characteristics are used to represent each tuple Y, and each tuple has N values. Assuming the number of classes are *x* and it labels from $C_1, C_2, C_3, \ldots\ldots C_m$ for any new tuple. Then the classifier will predict that the Y is a member of *m* class having the highest probabilty condition on Y.

$$\text{If P } (\frac{Ci}{X}) > P\left(\frac{Cj}{x}\right) where\ 1 \leq j \leq m.$$

The class Ci for which $P(\frac{Ci}{X})$ is maximized is called maximum posterior hypothesis.

As $P(X)$ is constant for all the classes it is not considered & the formulas becomes

$$P(\frac{Ci}{x}) = P\left(\frac{X}{Ci}\right) * P(ci)$$

In order to predict the class label of X, calculate $P\left(\frac{x}{ci}\right) * P(ci)$ is evaluated for each class Ci and the predictor class label is class Ci for which $P\left(\frac{x}{ci}\right) * P(ci)$ is maximum.

2.2.2 Data Source

For training data, 38 records with 7 medical attributes were used. For experimenting the data, excel is used straight. The software will read data from .csv files and pre-process it and then normalize the data-set for training and testing. For traing and testing the cross-validation will be 70% and 30%.

2.3 Current System Analysis

The system can predict more than one type of cancer or 7 types of cancers. There are two types of user in the current system which is the admin and patients. The functions of the system will be explained in terms of modules as below:

II) **Admin module**

Admin can add, modify or delete the training data. After adding the training data, other admins can view the data. Admin can access:

1. Upload training data
2. Add doctor's information
3. View patients' information
4. View Feedback
5. View doctors' information

II) **User module**

User can check the results of their cancer types**.** The following are the steps:

1. Registration
2. Check Cancer Symptoms by giving these input:
   - Age.
   - Gender.
   - Blood clots.
   - Urination.
   - Chest pain.
   - Coughing.
   - Pain/Itching.
   - Number of Red/white patch on the gums.
   - Memory diffuculty.

- Number of reddish, raised patch on skin.
- Digestion problem.
- Number of unexpected or sudden weight loss.

- The system will lead patients to the doctor's consultation page.
- Provide comments: Patients can write comments about the system.
- View doctor : Patients will be redirected to relevant physicians depending on the sort of cancer that has been predicted.

2.4 Comparison between existing systems

The invoicing of patients, inventory management, and the creation of basic statistics are all supported by several hospital information systems. There are some decision support systems in place in certain hospitals, but they are rather restricted.

But, also now a days there are also many intelligent systems like the current system analysis using machine learning approach. But the main drawback is that those systems are difficult to maintain. It is less user-friendly. And most importantly, it is only made for predicting specific task only, for example, predict only one type of cancer. Almost all the existing system can only predict one type of cancer. And the system is difficult to maintain because the training data needs to be updated at least every few months because everyday new data is coming and it needs to upload into the database manually whereas in the current system, the admin login into the system and upload the data from the web application and the system automatically update all the training data which means it consumes less time for processing the activities.

The current system is very user-friendly because once the predicted cancer type result is out, the system will automatically show all the relevant doctors to consult and also ask the patients to provide feedback regarding about the system. Depending on the feedback, later any update can be made for the system to make it more user-friendly.

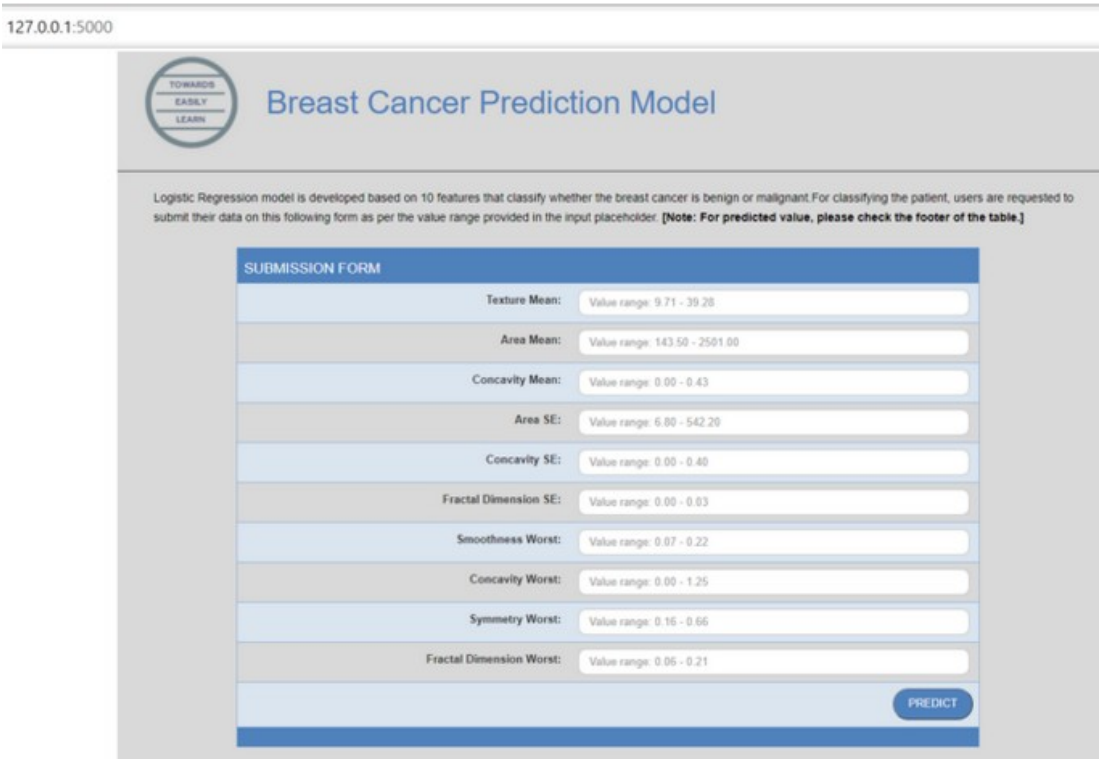


Figure 2 Existing System only specific for breast cancer

Figure 3 Existing System only specific for lung cancer and not user-friendly

2.5 Literature Review of Technology Used

The Project is loaded in Visual Studio 2022. Visual Studio is used for Design and coding of project. Created and maintained all databases into SQL Server 2010 using SQL Management Studio for creating tables, writing query for store data or record of project.

Front End Technology

Microsoft .NET Framework

The.NET Framework is a computer environment which makes it easier to construct applications for the Internet's widely dispersed users. The following goals are the focus of the .NET framework:

- To offer a uniform object-oriented programming domain regardless of whether object code is stored and run locally, locally but disseminated via the Internet, or remotely.
- Versioning control history and software deployment can be decreased by provding plattoforms like Jira or Jenkins.
- Code-execution environment assures the safe execution of code, even when written by a third party who is unknown or only partially trustworthy.
- Offers code to be executed without the performance issues of scripted or interpreted domains being present.
- Offers an uniform developer experience over a broad range of application variantions, including windows-based and web-based application.

- To confirm that code written in the .NET framework may be easily combined with other code written in any other programming language.

2.6 Chapter Summary

The chapter describes about the proposed system. It describes in detail how the classifier is splits the training and testing data and evaluating the model for prediction. The source of data is collected from World Health Organization an feed into the Naive Bayes algorithm classifier. After classification, the proposed system automatically redirects user to view doctors.

This chapter also talks about the related works have been done before on this system and also describes why this system is better choice than any other prediction system.

Lastly, in this chapter describes how the technology is used to build his project. This project used .NET framework, object-oriented techniques and SQL Management Studio 2010. It also describes why this technology has chosen.

3

SYSTEM DEVELOPMENT METHODOLOGY

3.1 Introduction

System development methodology is a a framework that follows the standard process of a project's structure, plan, control, organization and the development of an information system. By defining the tasks to be performed and the methods to be applied, system development methodologies are helps for an example, to enhance the management and control of the software life cycle. Additionally, it standardises the development process and end result while streamlining and structuring the process. The employment of a system development methodology is often implicitly expected to increase the productivity and quality of system development.

3.2 Methodology Choice and Justification

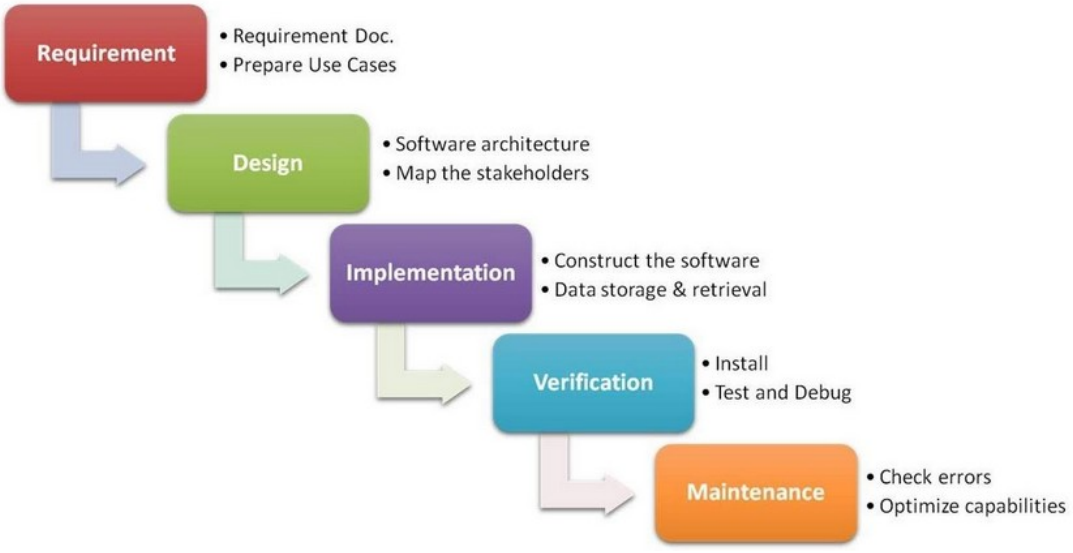


Figure 4 Waterfall Methodology

One of the main reasons for choosing the waterfall model is becasue it works pretty well for smaller projects when requirements are quite clear and understandable. This project is not for the purpose of running on a large scale or commercial uses. For smaller project like this, waterfall model is simple and easy to implement. Every phases from Requirements to Maintenances are completed one by one before moving to another phases. Therefore, if any issues or problem arise from any phases, it can be detected for early recovery. Thus, it is very simple to manage as the development cycle is linear and sequential.

3.3 Phases of the Chosen Methodology

REQUIREMETS – Software Requirements Specifications with Use Cases have been developed for this phase. In SRS, all the activitiy and sequence diagrams have been drawn and elicited from the requirements.

DESIGN – Software Design Documents have been developed for this phase. Software Architecural Model has been proposed for the system, Database has been designed and the flow of events have been spicified in the SDD.

IMPLEMENTATION – Later in the Final Year Project 2, all the architecture have been developed in SDD will be implemented and the full software will be contructured.

VERIFICATION – For verification and validation purposes, the way how the system should validate all the input and output flow are discussed in the Software Test Documentation with Test Approaches and Test Cases.

MAINTENANCE – Later in FYP2, when the software will be completely developed, then the application will be check for any errors or no.

3.4 Technology Used Description

Project Implementation

The Project is loaded in Visual Studio 2010. Visual Studio is used for Design and coding of project. Created and maintained all databases into SQL Server 2010, in that where create tables, write query for store data or record of project.

I. Hardware Requirement:

- Processor: Core i3 or higher
- Memory: 1 GB RAM or higher
- Hard Disk Drive: 50 GB
- Monitor: Any LCD, or LED
- Internet: Should be connected

II. Software Requirement:

- Operating System: Windows Vista, 7 or higher version
- IDE: Microsoft Visual Studio 2010 or higher (Community Enterprise)
- IDE for Database: Microsoft SQL Server 2010 or higher.

Front-End Technology – Microsoft .NET Framework (View)
Middleware Technology – Active Data Objects.Net (Controller)
Backned Technology – Microsoft SQL Server (Model)

3.5 System Requirement Analysis

Functional Requirements:

I. Allows the admin to login and view or add training data for the algorithm to analyze and predict results.

II. Allows the admin to add or view specialist details with respect to types of cancer.

III. Allows the admin to view all the registered users' details and their feedback.

IV. Allows the user to register with their basic details and predict cancer by inputting all the necessary details and the system will accordingly view doctor to consult.

V. Allow the user to view various doctor based on the predicted cancer.

Non-Functional Requirements:

I. The system should be able to authorize the admin and patients via username and password.

II. The user manual should struct clearly with example. The manual should explain to the user what the software should do and how it functions. The manual is used as a reference so that the user can understand the system easily.

III. The system shall allow the users to access the system from any device that has internet connection and internet browsing.

IV. The information of the users are only able to be accessed by authority that is authorized by the system.

V. The system shall support up to 1000 users at the same time with loading time is at most 8.5 to less than 5 seconds.

3.6 Chapter Summary

The purpose of this chapter is to find out how the chosen methodology is used in this project. The specific concerns covered include the types of methodologies being employed, the degree of adherence to the methodology's specifics, and the integration of those phases with their methodologies.

This chapter also discusses what technology is used to build the proposed architecture including the front-end and back-end technologies using MVC pattern. Lastly, the chapter discusses all the functional and non-functional requirements that has been elicited for system requirements analysis.

4

REQUIREMENT ANALYSIS AND DESIGN

4.1 Introduction

Requirements analysis and design is the process of gathering all information, modeling the requirements, designing and validating of all information. It finds the solution of business requirements through elicitation. After eliciting all the requirements, then it needs to work on requirements analysis and design.

The important elements of RAAD is to model the requirements, analyze the requirements, represent the requirements and implement the requirements' appropriate levels of abstraction. All these tasks are done by involving the stakeholders because the requirements need to approve by both analyst and stakeholders. All the requirements and design should be as clear as crystal in order to understand the model that follows the standards.

This project is a final year project where all the requirements are elicited from the previous existing system. It does involve stakeholders like doctors or patients, but the requirements are not elicited from them. This system is developed by looking at the other existing system and modifying the existing system in order for making it more ease for both patients and doctors as well as the admin who will be handling the system.

4.2 Requirement Analysis

The system features include user registration, log-in, check results, view doctor, view/provide feedback, add training data, view user details, view training data, provide feedback.

Figure 5 Use Case Diagram of <Cancer Prediction System>

Domain Model

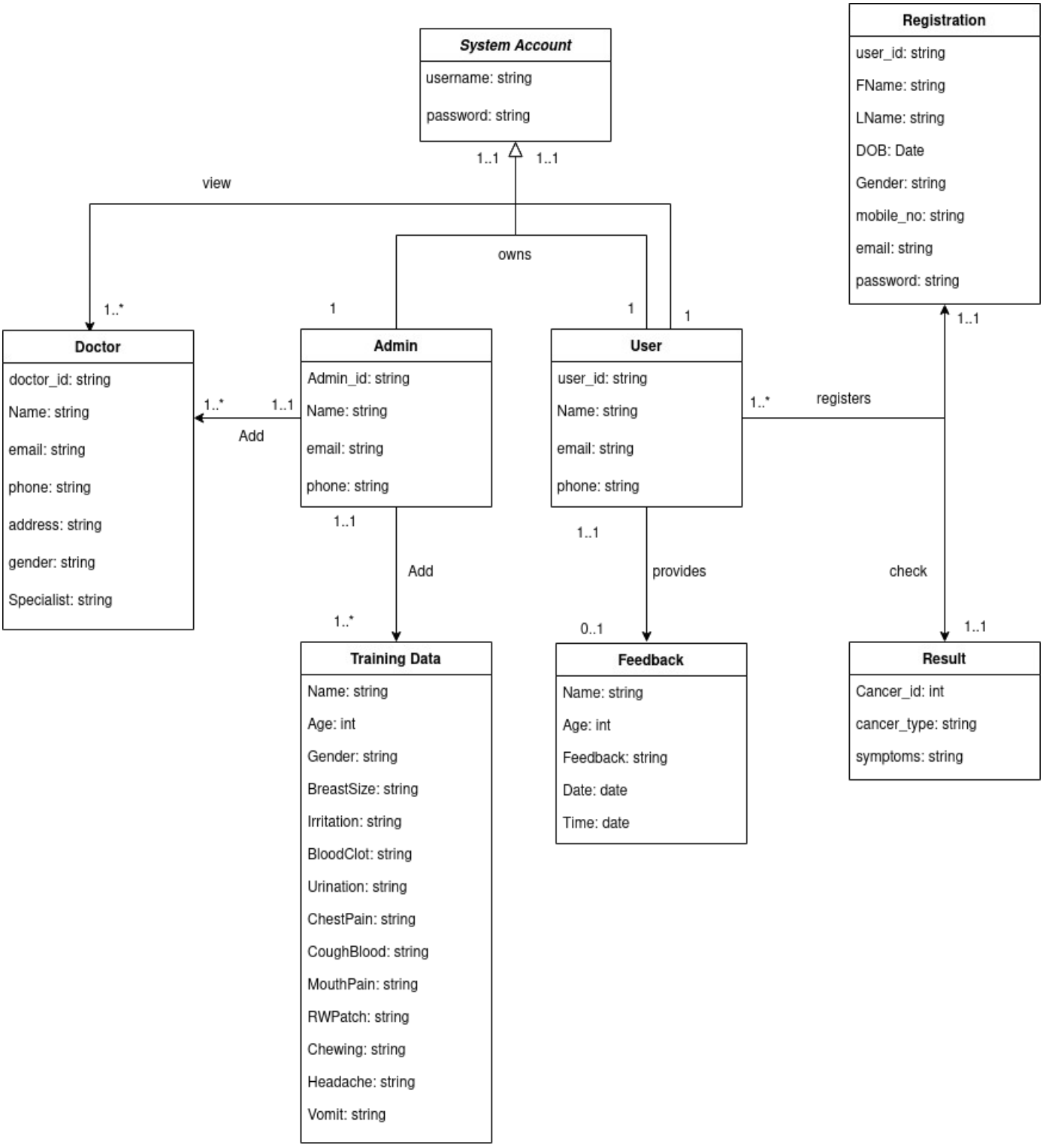


Figure 6 Domain Model of <Cancer Prediction System>

State-based Transition

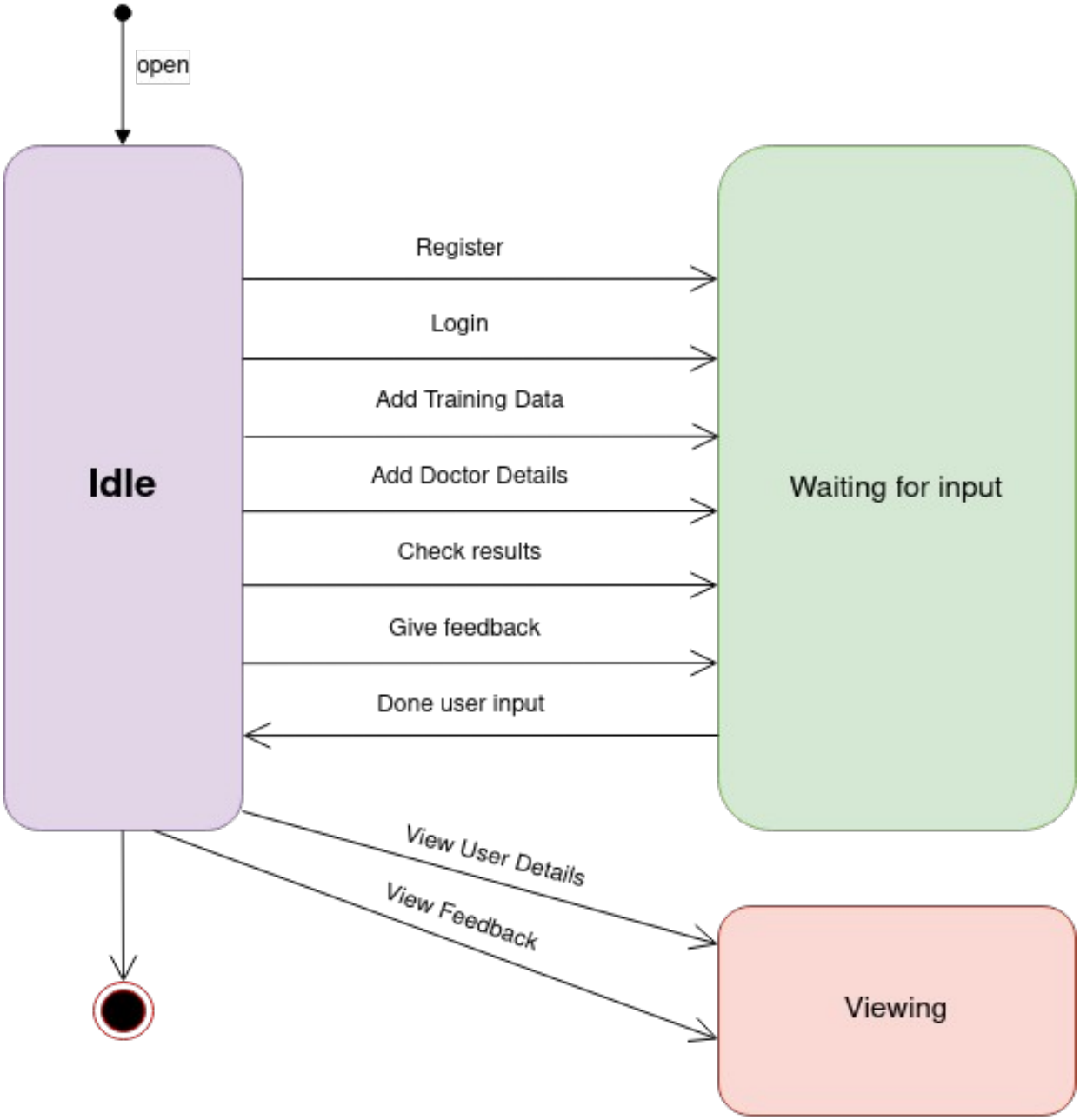


Figure 7 State Machine Diagram of the Cancer Prediction System

4.3 Project Design

System Architectural Design

Architecture Style and Rationale

The chosen architectural design pattern is model view controller (MVC). The main reason that cause to choose this model is due to the system has multiple ways to view and interact with data and the future requirements for interaction and presentation of data are unknown. MVC is an acronym for model, view and controller and it is a product development architecture. By using MVC, it could create application that separate the different aspects of the application. Since MVC allows the data to change independently of its representation and vice versa, it enables to focus on one aspect of the implementation at a time. For example, it can focus on the controller(input logic) without depending on models and views (business logic and UI logic). By using MVC it allows to modify the system easily and multiple developers can work simultaneously on the model, controller and views.

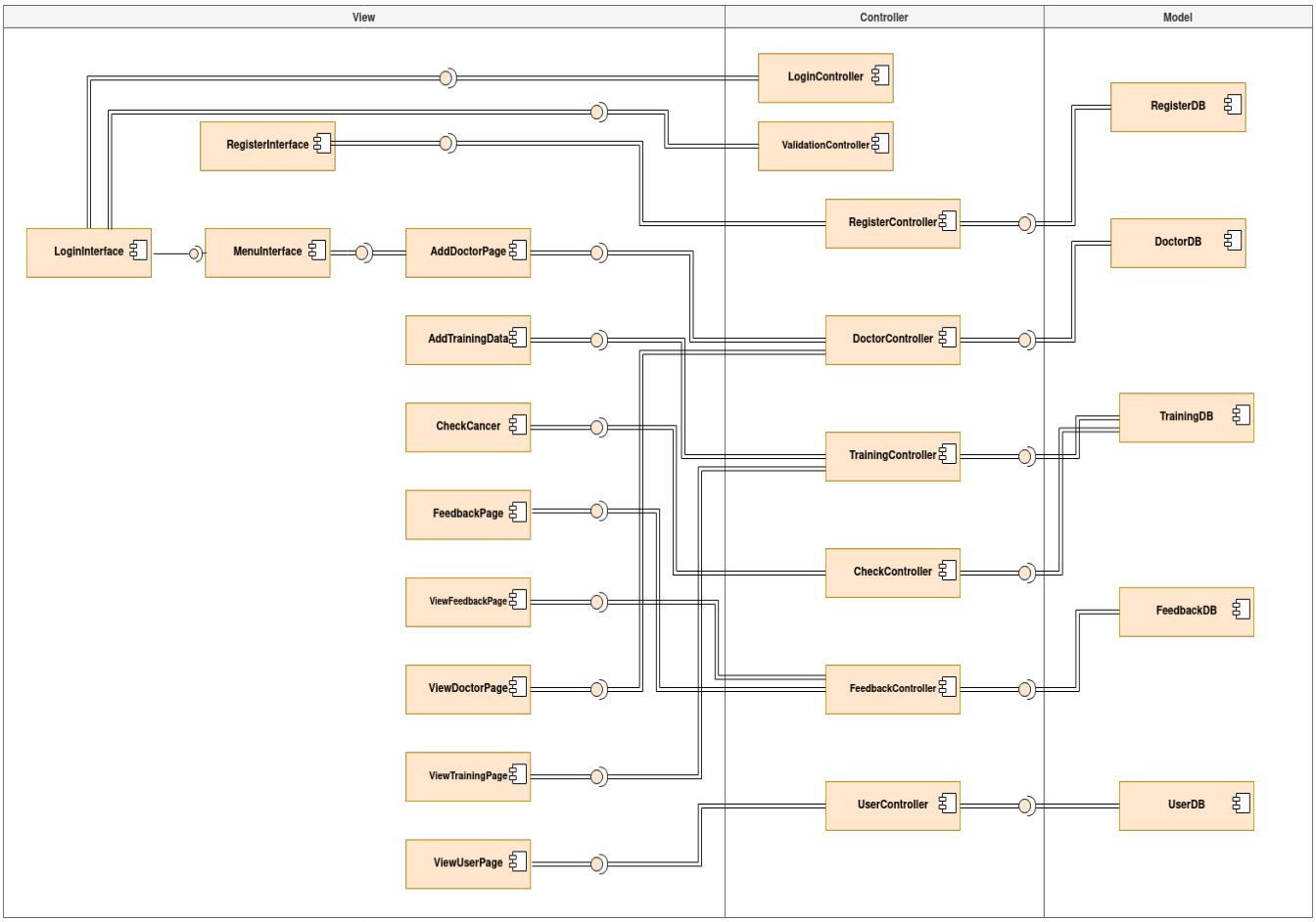


Figure 8 Component Model of <Cancer Prediction System>

Complete Package Diagram

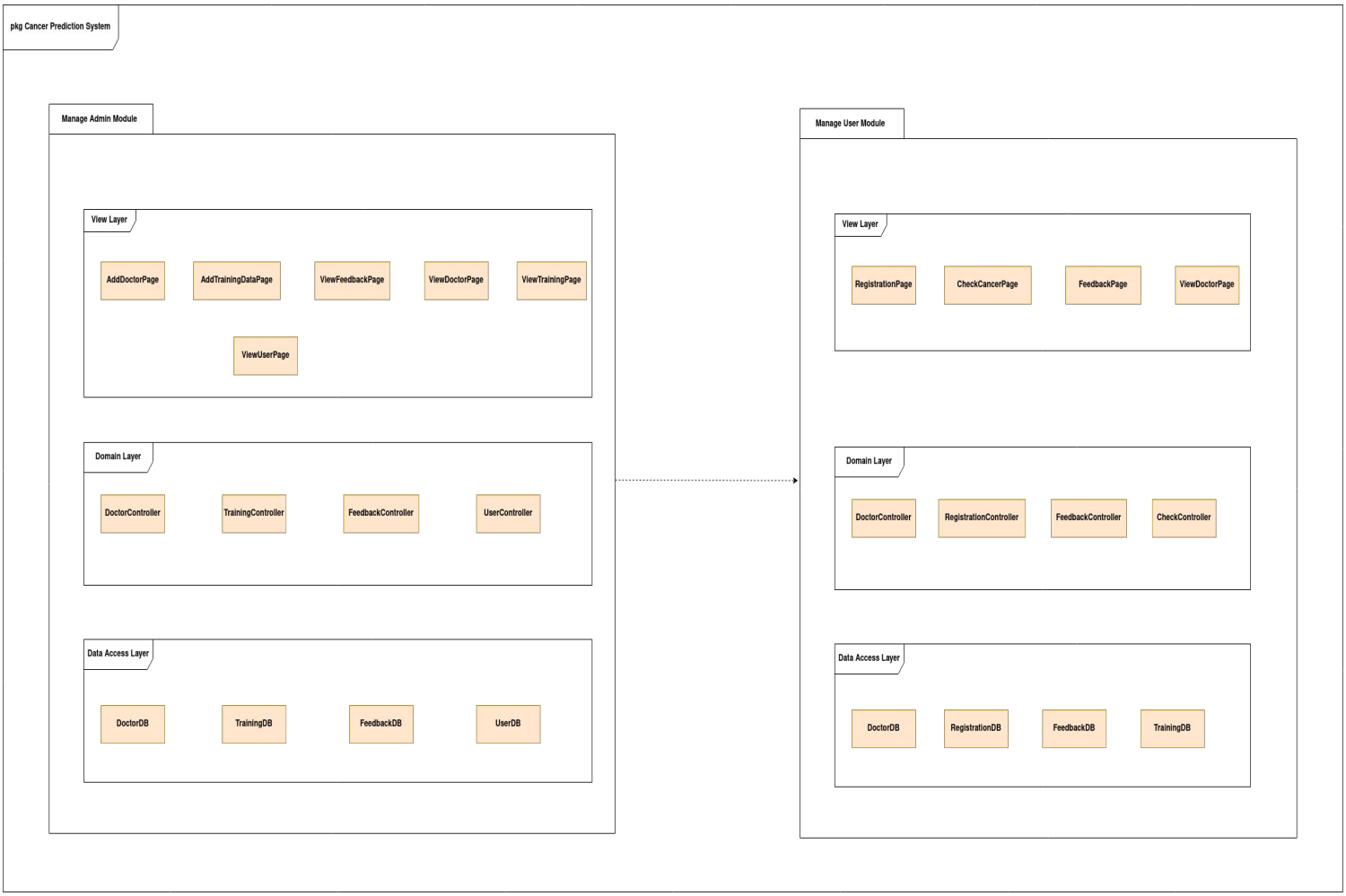


Figure 9 Complete Package Diagram

Overall Class Diagram

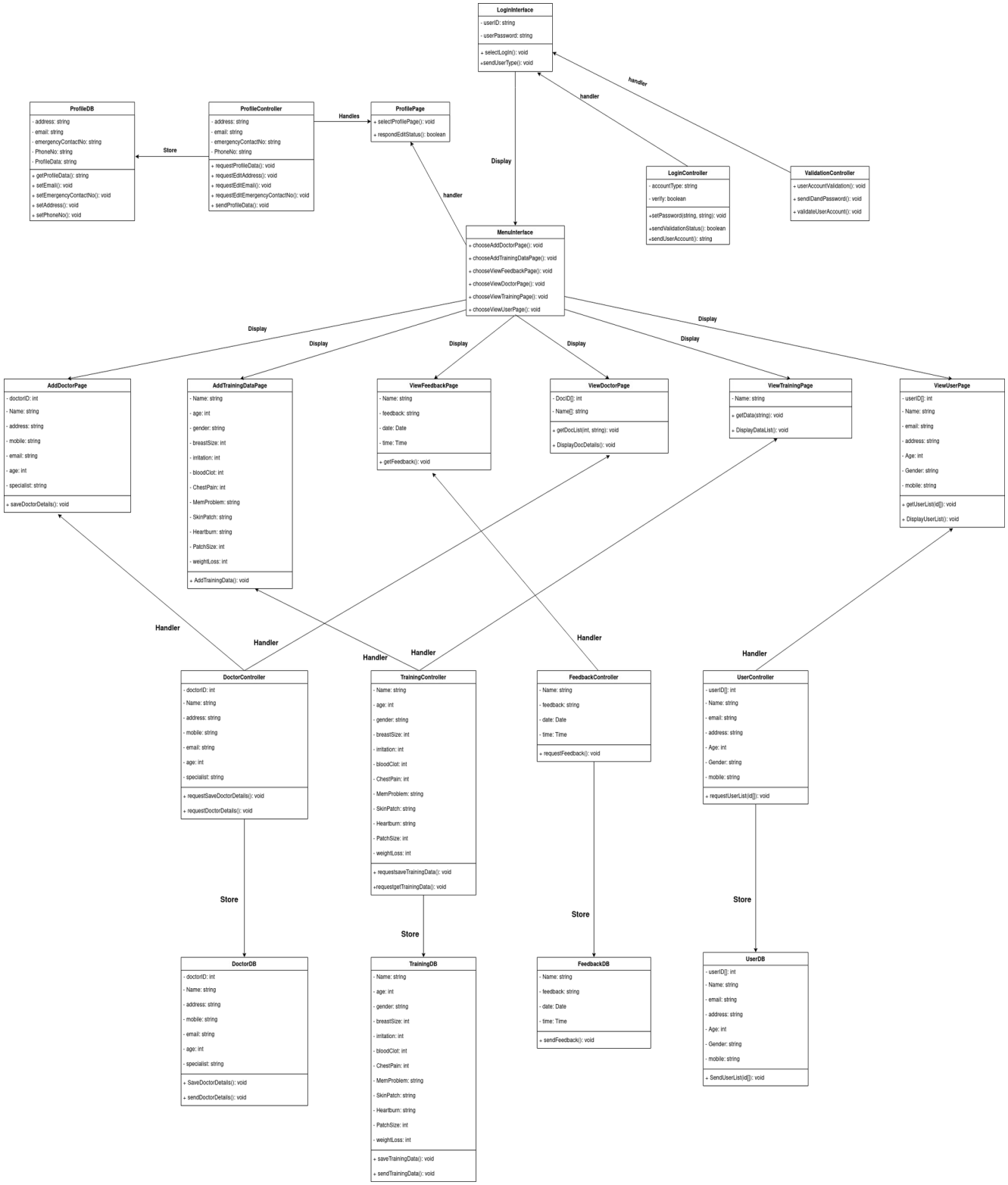
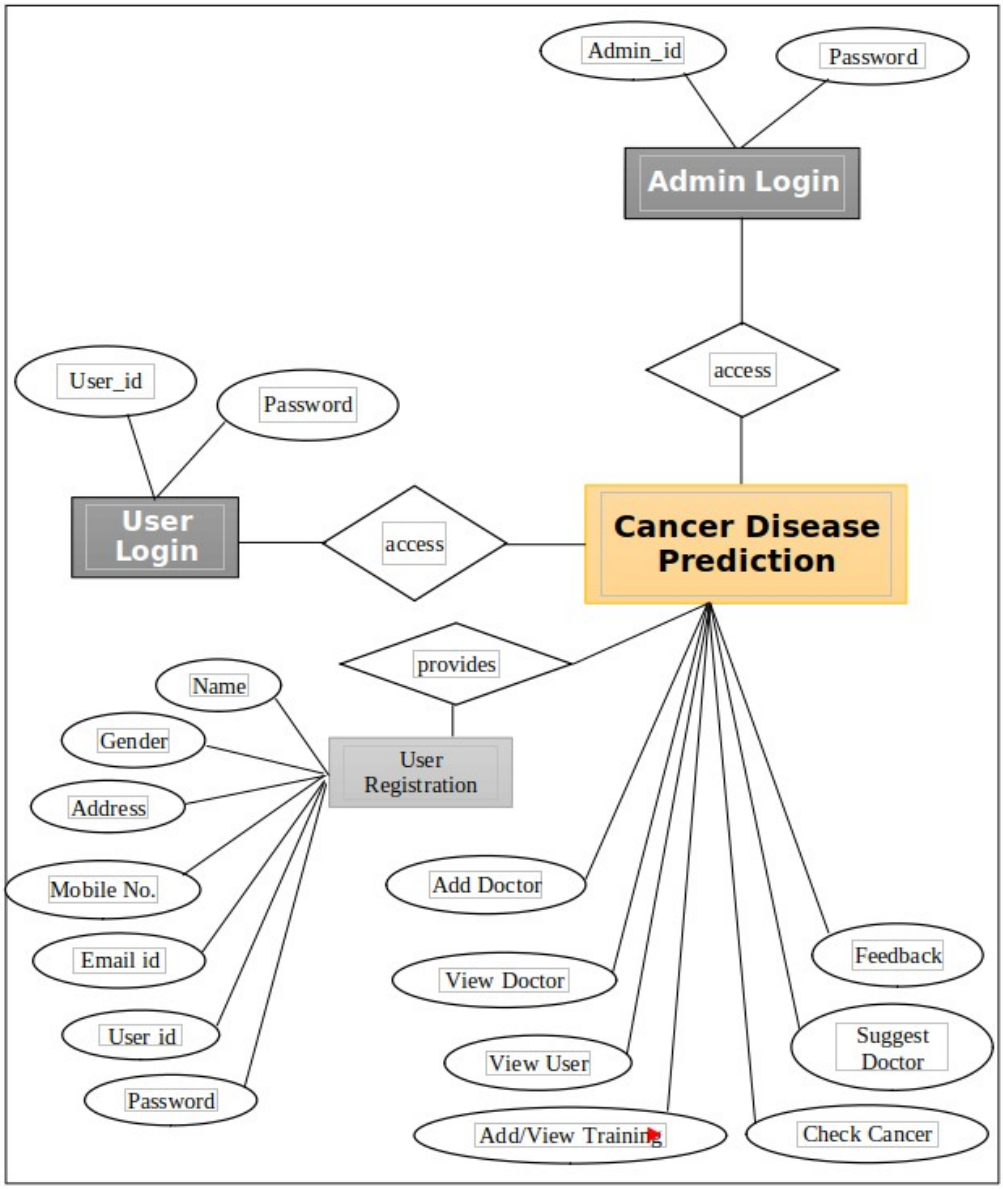


Figure 10 Overall Class diagram

4.4 Database Design

The major data or systems entities are stored into a relational database named as “CancerPrediction” database and comprise of 5 entities (for manage admin module subsystem and user module subsystem). The data retrieved or stored by system though the controllers from the entities or to the entities. The data is organized into attributes with the recognizable names. Each of the entities has its own contents and all of these entities have different unique name. Each of these unique names enable the system search and retrieve the data correctly when needed. Each of the entities consists its own attributes.

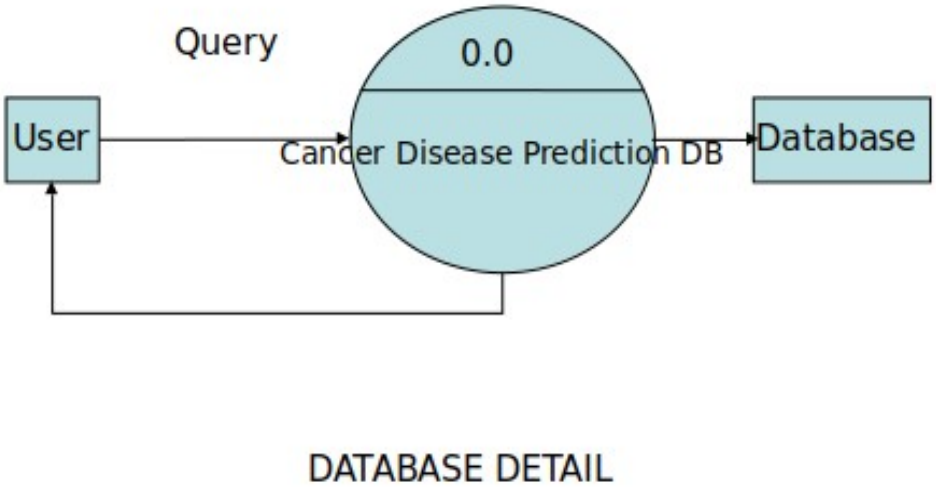
Entity Name	**Description**
DoctorDB	This entity stores all the details of the doctor that has been registered by the admin.
FeedbackDB	This entity consists of all the reviews and feedback shared by the patients and can be viewed by the admin.
TrainingDB	This entity consists of all the training data uploaded by the admin to train the machine learning model.
UserDB	This entity consist of all the details of the patients’ input for predicting the cancer type. The admin can view all this data input by the patients.
RegisterDB	This entity stores all the basic information of the user after the user finished regsitration, for example, name, email, password.



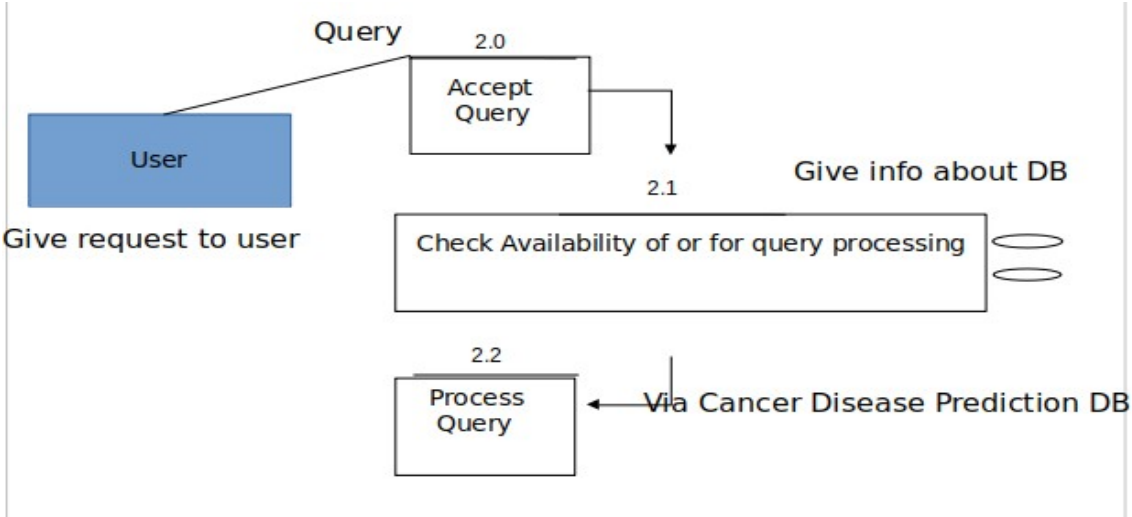
Entity-Relationship Diagram

Figure 11 ERD diagram for Patient(User) and System Administrator

DFD Diagrams



DFD Level 1 – Checking Result for cancer type



DFD Level 2: Prediction of cancer

Implementing in SQL

*CancerPredict.sql ×

```
USE [CancerPrediction]
GO
/****** Object:  Table [dbo].[DocDetails]    Script Date: 2021/06/24 02:07:36 PM ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
SET ANSI_PADDING ON
GO
CREATE TABLE [dbo].[DocDetails](
    [Did] [varchar](50) NULL,
    [Name] [varchar](50) NULL,
    [Address] [varchar](50) NULL,
    [Mobile] [varchar](50) NULL,
    [Email] [varchar](50) NULL,
    [Age] [varchar](50) NULL,
    [Gender] [varchar](50) NULL,
    [Specialist] [varchar](50) NULL
) ON [PRIMARY]
```

Figure 12 DoctorDB

```
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
SET ANSI_PADDING ON
GO
CREATE TABLE [dbo].[Feedback](
    [Name] [varchar](50) NULL,
    [Feedback] [varchar](max) NULL,
    [Date] [varchar](50) NULL,
    [Time] [varchar](50) NULL
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]

GO
SET ANSI_PADDING OFF
GO
/****** Object:  Table [dbo].[Register]
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
SET ANSI_PADDING ON
GO
CREATE TABLE [dbo].[Register](
    [Id] [varchar](50) NULL,
    [Name] [varchar](50) NULL,
    [Address] [varchar](max) NULL,
    [Mobile] [varchar](50) NULL,
    [Email] [varchar](50) NULL,
    [Age] [varchar](50) NULL,
    [Gender] [varchar](50) NULL,
    [Password] [varchar](50) NULL
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
```

Figure 13 FeedbackDB and RegisterDB

```
GO
/****** Object:  Table [dbo].[TrainingData]
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
SET ANSI_PADDING ON
GO
CREATE TABLE [dbo].[TrainingData](
    [Name] [varchar](50) NULL,
    [Age] [varchar](50) NULL,
    [Gender] [varchar](50) NULL,
    [BreastSize] [varchar](50) NULL,
    [Irritation] [varchar](50) NULL,
    [BloodClot] [varchar](50) NULL,
    [Urination] [varchar](50) NULL,
    [ChestPain] [varchar](50) NULL,
    [CoughBlood] [varchar](50) NULL,
    [MouthPain] [varchar](50) NULL,
    [RWPatch] [varchar](50) NULL,
    [Chewing] [varchar](50) NULL,
    [Headache] [varchar](50) NULL,
    [Vomit] [varchar](50) NULL,
    [MemProblem] [varchar](50) NULL,
    [SkinPatch] [varchar](50) NULL,
    [PatchSize] [varchar](50) NULL,
    [Heartburn] [varchar](50) NULL,
    [StomachBloat] [varchar](50) NULL,
    [ShoulderPain] [varchar](50) NULL,
    [WeightLoss] [varchar](50) NULL
) ON [PRIMARY]
```

Figure 14 TrainingDB

```
/****** Object:  Table [dbo].[UserData]
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
SET ANSI_PADDING ON
GO
CREATE TABLE [dbo].[UserData](
    [Name] [varchar](50) NULL,
    [Age] [varchar](50) NULL,
    [Gender] [varchar](50) NULL,
    [BreastSize] [varchar](50) NULL,
    [Irritation] [varchar](50) NULL,
    [BloodClot] [varchar](50) NULL,
    [Urination] [varchar](50) NULL,
    [ChestPain] [varchar](50) NULL,
    [CoughBlood] [varchar](50) NULL,
    [MouthPain] [varchar](50) NULL,
    [RWPatch] [varchar](50) NULL,
    [Chewing] [varchar](50) NULL,
    [Headache] [varchar](50) NULL,
    [Vomit] [varchar](50) NULL,
    [MemProblem] [varchar](50) NULL,
    [SkinPatch] [varchar](50) NULL,
    [PatchSize] [varchar](50) NULL,
    [Heartburn] [varchar](50) NULL,
    [StomachBloat] [varchar](50) NULL,
    [ShoulderPain] [varchar](50) NULL,
    [WeightLoss] [varchar](50) NULL,
    [Result] [varchar](max) NULL
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
```

Figure 15 UserDB for data mining and prediction

Fig 12 shows the database of all the details of a doctor.

Fig 13 shows the database of feedback given by the user and user's registration data.

Fig 14 shows the database of all the training data for classification of cancer types.

Fig 15 shows the database of user's data for predicting the cancer type.

4.5 Interface Design

For manage admin module subsystem, after the admin login their account, the system will display a menu that only shows functions that is relevant to the admin. For admin, the menu will show five functions such as Add training data, Add doctor details, View feedback, View User details and view training data.

For Add training data, the system will display a add new training page for the admin to upload spreadsheet files. For Add doctor details, the system will display the admin to fill up all the details and upload the resume for the doctors. After uploading, the patients can be able to see the doctors based on their predicted cancer type. For View feedback, the system will display all the reviews given by the patients. For user details, the system will display all the list of registered users. For view training data, the system will display to the admin all the list of data files uploaded.

Screen images

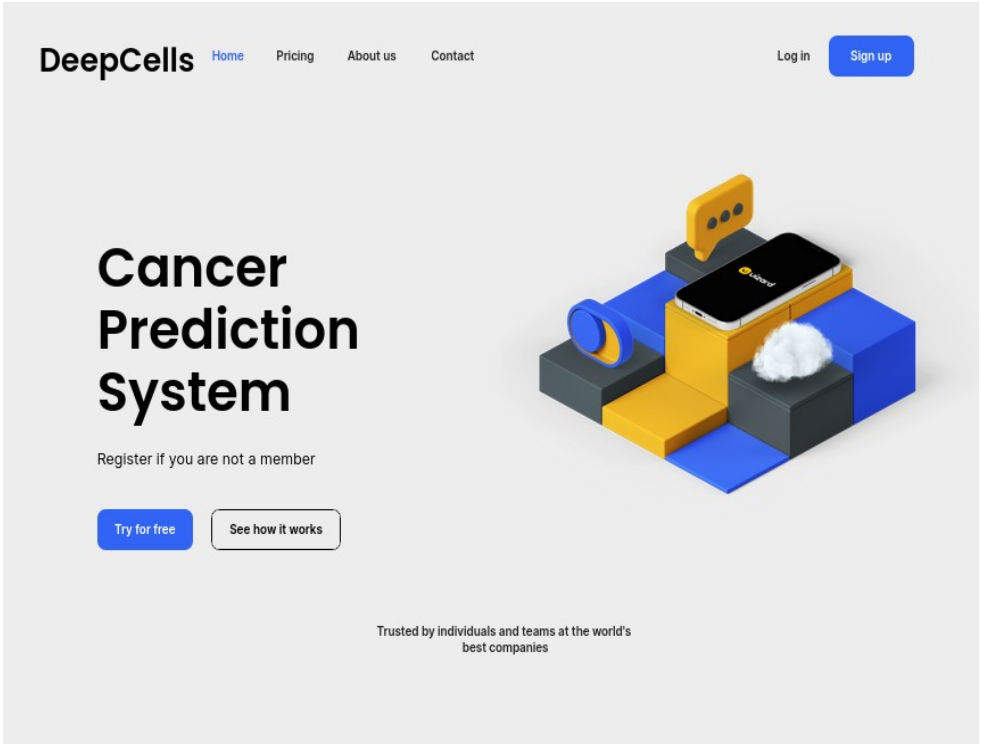


Figure 16 Homepage Interface

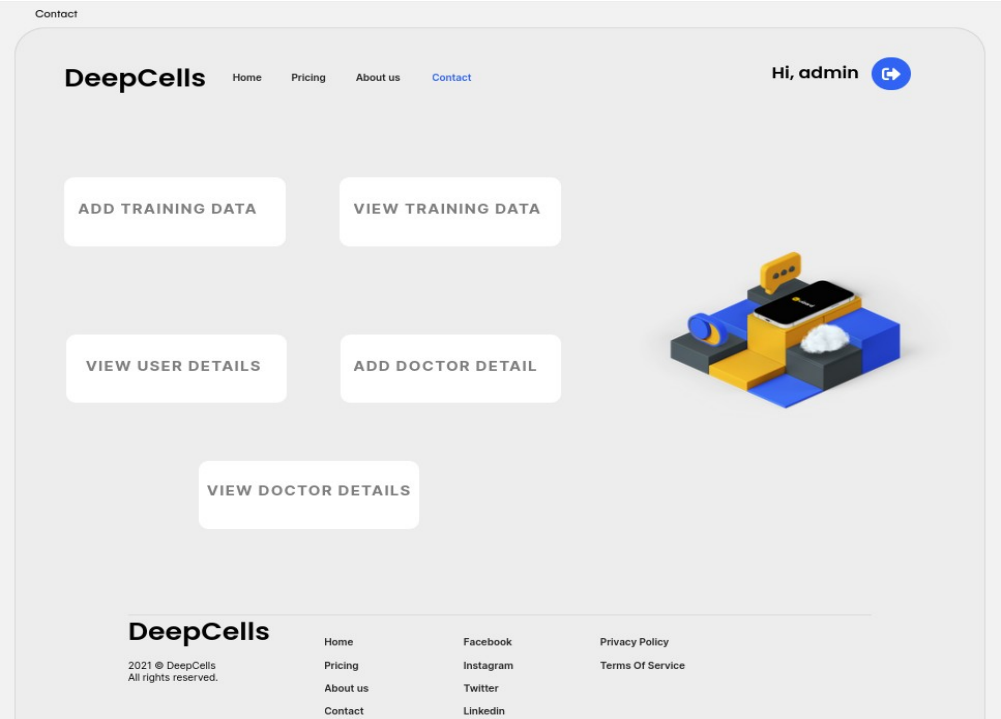


Figure 17 Menu Interface for Admin

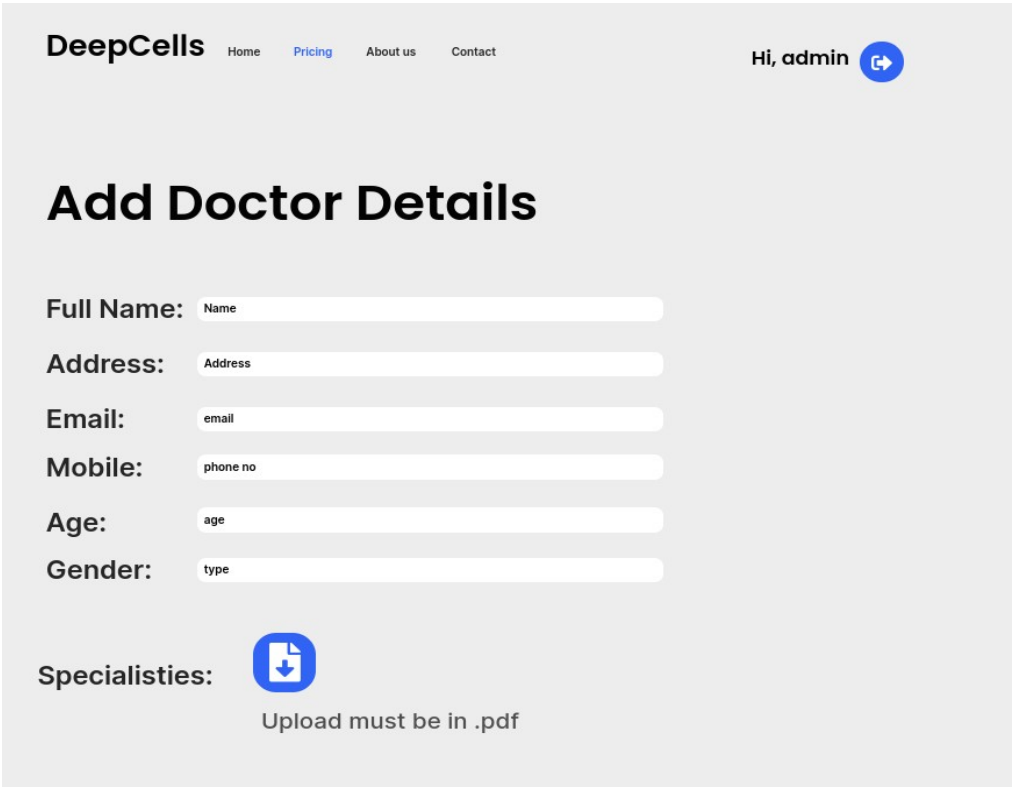


Figure 18 Add Doctor Details interface

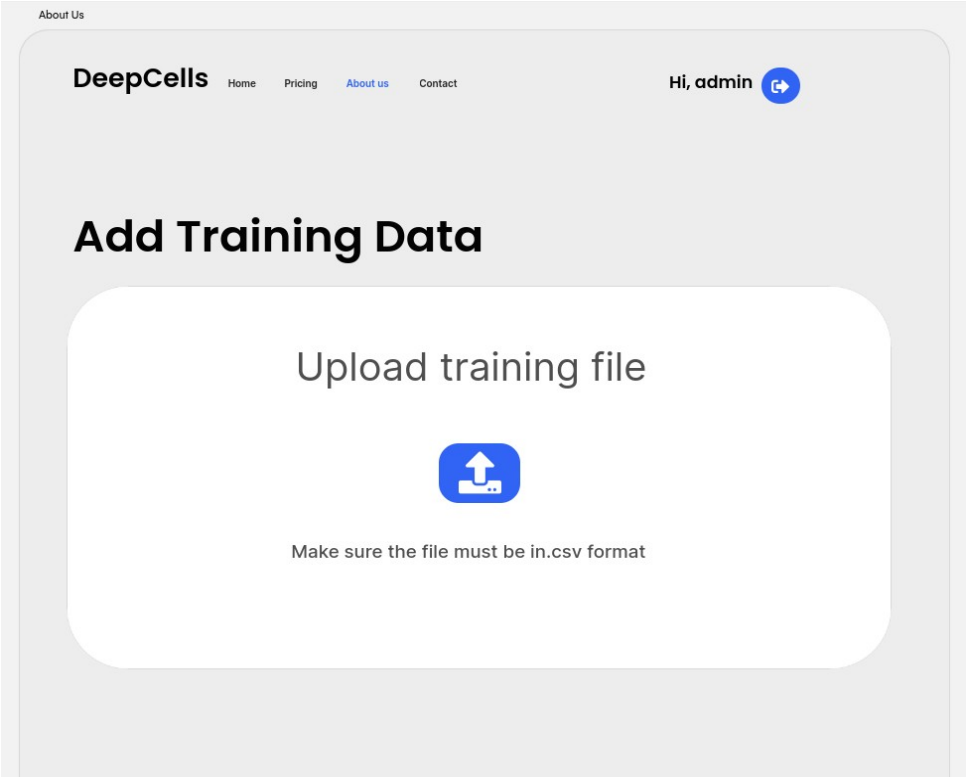


Figure 19 Add Training Data

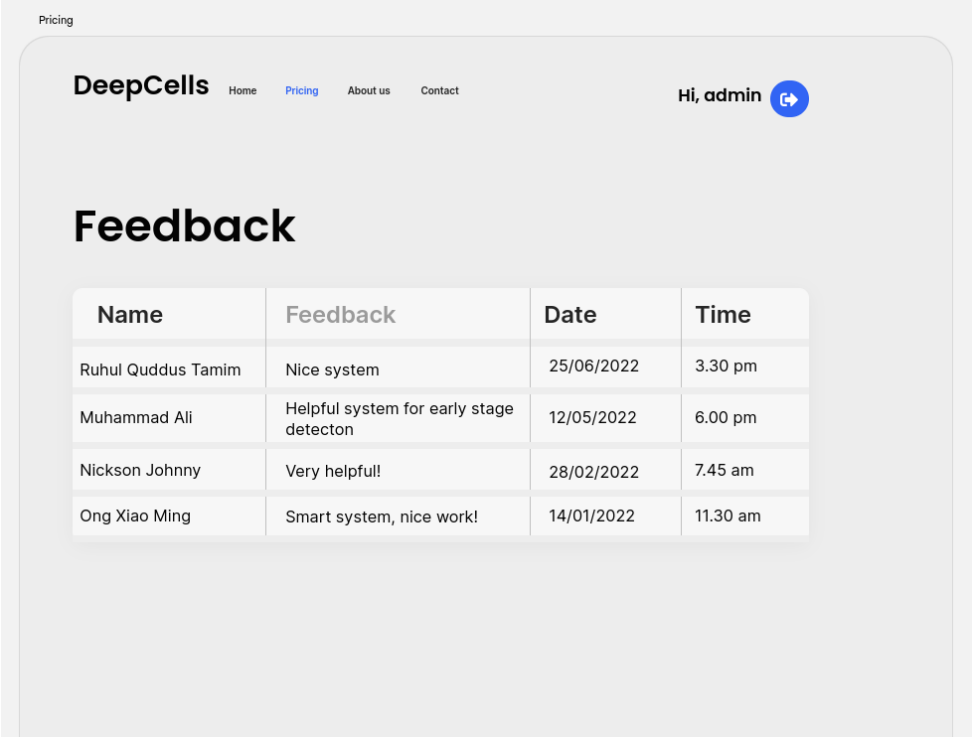


Figure 20 Feedback by patients

4.6 Chapter Summary

This chapter describes the Cancer Prediction System which is to make a new version of the current system. It introduces the requirements analysis and design, and also shows that how the requirements has been elicited from the previous existing system and prepared the Use-Cases, domain model and state-based transition diagrams.

This chapter also discusses how to design the project from a very detailed design to database design. First, it describes the design pattern which is MVC, then the system divides into two subsystems (User Module and Admin Module). From the design pattern to overall class diagram. Then, the database is design and implemented in SQL. Lastly, it shows the prototype of few screen images to show how the system should look like.

5

IMPLEMENTATION AND TESTING

5.1 Introduction

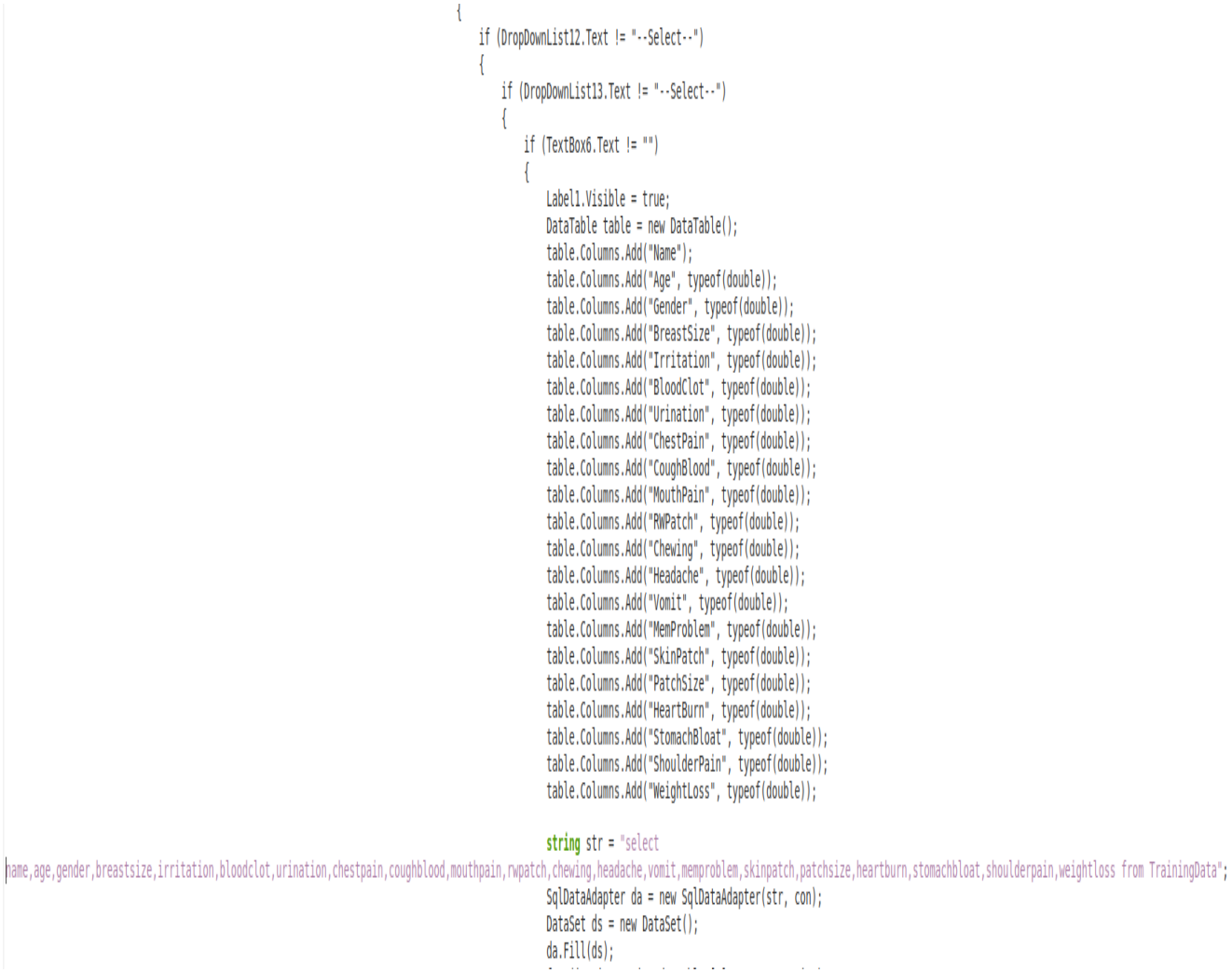
The implementation phase is the most critical step in the software development process. At this step, the system's physical source code has been generated. Programmers develop the IT system based on the requirements obtained and the project documentation provided.

5.2 Coding of System Main Functions

One of the most main function in the system for this project is checking or predicting the cancer type. To calculate the cancer type, it uses naive Bayes classifier functions. The function is called from different folder in the project. After calling the functions, the classifier calls another function called “Helper” functions to pre-process the data before training and predicting the cancer type from the user input. Below are the few images to show the code part of the checking cancer type UI pages, classifier function and “Helper” function.

```
using System;
using System.Collections.Generic;
using System.Configuration;
using System.Data;
using System.Data.SqlClient;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.HtmlControls;
using System.Web.UI.WebControls;

public partial class Check : System.Web.UI.Page
{
    SqlConnection con = new SqlConnection(ConfigurationManager.ConnectionStrings["ConString"].ConnectionString);
    protected void Page_Load(object sender, EventArgs e)
    {
        LabelErr18.Visible = false;
        LabelErr.Visible = false;
        LabelErr19.Visible = false;
        LabelErr0.Visible = false;
        LabelErr1.Visible = false;
        LabelErr2.Visible = false;
        LabelErr3.Visible = false;
        LabelErr4.Visible = false;
        LabelErr5.Visible = false;
        LabelErr6.Visible = false;
        LabelErr7.Visible = false;
        LabelErr8.Visible = false;
        LabelErr9.Visible = false;
        LabelErr10.Visible = false;
        LabelErr11.Visible = false;
        LabelErr12.Visible = false;
        LabelErr13.Visible = false;
        LabelErr14.Visible = false;
        LabelErr15.Visible = false;
        LabelErr16.Visible = false;
        LabelErr17.Visible = false;
        LabelErr5.Visible = false;
        //string Uid = Session["Uid"].ToString();
        //string str = "select age,gender from register where id='" + Uid + "'";
        //SqlDataAdapter da = new SqlDataAdapter(str, con);
        //DataSet ds = new DataSet();
        //da.Fill(ds);
        //string age = ds.Tables[0].Rows[0][0].ToString();
        ////lblAge.Text = age;
        //string sex = ds.Tables[0].Rows[0][1].ToString();
        //if(sex=="Male")
        //{
        //    //lblSex.Text = "1";
        //}
        //else
```

Implementation of UI page for main function



```
{
    if (DropDownList12.Text != "--Select--")
    {
        if (DropDownList13.Text != "--Select--")
        {
            if (TextBox6.Text != "")
            {
                Label1.Visible = true;
                DataTable table = new DataTable();
                table.Columns.Add("Name");
                table.Columns.Add("Age", typeof(double));
                table.Columns.Add("Gender", typeof(double));
                table.Columns.Add("BreastSize", typeof(double));
                table.Columns.Add("Irritation", typeof(double));
                table.Columns.Add("BloodClot", typeof(double));
                table.Columns.Add("Urination", typeof(double));
                table.Columns.Add("ChestPain", typeof(double));
                table.Columns.Add("CoughBlood", typeof(double));
                table.Columns.Add("MouthPain", typeof(double));
                table.Columns.Add("RWPatch", typeof(double));
                table.Columns.Add("Chewing", typeof(double));
                table.Columns.Add("Headache", typeof(double));
                table.Columns.Add("Vomit", typeof(double));
                table.Columns.Add("MemProblem", typeof(double));
                table.Columns.Add("SkinPatch", typeof(double));
                table.Columns.Add("PatchSize", typeof(double));
                table.Columns.Add("HeartBurn", typeof(double));
                table.Columns.Add("StomachBloat", typeof(double));
                table.Columns.Add("ShoulderPain", typeof(double));
                table.Columns.Add("WeightLoss", typeof(double));

                string str = "select
name,age,gender,breastsize,irritation,bloodclot,urination,chestpain,coughblood,mouthpain,rwpatch,chewing,headache,vomit,memproblem,skinpatch,patchsize,heartburn,stomachbloat,shoulderpain,weightloss from TrainingData";
                SqlDataAdapter da = new SqlDataAdapter(str, con);
                DataSet ds = new DataSet();
                da.Fill(ds);
```

Implementation of UI page for main function

```
da.Fill(ds);
for (int i = 0; i < ds.Tables[0].Rows.Count; i++)
{
    table.Rows.Add(ds.Tables[0].Rows[i][0].ToString(),
    Convert.ToDouble(ds.Tables[0].Rows[i][1].ToString()),
    Convert.ToDouble(ds.Tables[0].Rows[i][3].ToString()),
    Convert.ToDouble(ds.Tables[0].Rows[i][4].ToString()),
    Convert.ToDouble(ds.Tables[0].Rows[i][5].ToString()),
    Convert.ToDouble(ds.Tables[0].Rows[i][6].ToString()),
    Convert.ToDouble(ds.Tables[0].Rows[i][7].ToString()),
    Convert.ToDouble(ds.Tables[0].Rows[i][8].ToString()),
    Convert.ToDouble(ds.Tables[0].Rows[i][9].ToString()),
    Convert.ToDouble(ds.Tables[0].Rows[i][10].ToString()),
    Convert.ToDouble(ds.Tables[0].Rows[i][11].ToString()),
    Convert.ToDouble(ds.Tables[0].Rows[i][12].ToString()),
    Convert.ToDouble(ds.Tables[0].Rows[i][13].ToString()),
    Convert.ToDouble(ds.Tables[0].Rows[i][14].ToString()),
    Convert.ToDouble(ds.Tables[0].Rows[i][15].ToString()),
    Convert.ToDouble(ds.Tables[0].Rows[i][16].ToString()),
    Convert.ToDouble(ds.Tables[0].Rows[i][17].ToString()),
    Convert.ToDouble(ds.Tables[0].Rows[i][18].ToString()),
    Convert.ToDouble(ds.Tables[0].Rows[i][19].ToString()),
    Convert.ToDouble(ds.Tables[0].Rows[i][20].ToString()));
}

Classifier classifier = new Classifier();
classifier.TrainClassifier(table);
string ans = "";
try
{
    ans = classifier.Classify
        (new double[]
        {
```

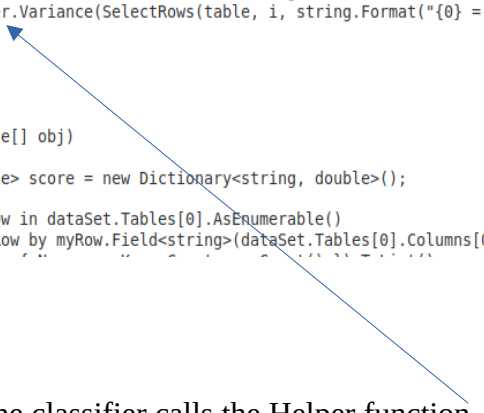
```
Convert.ToDouble(TextBox7.Text),
Convert.ToDouble(DropDownList1.SelectedValue.ToString()),
Convert.ToDouble(DropDownList2.SelectedValue.ToString()),
Convert.ToDouble(DropDownList3.SelectedValue.ToString()),
Convert.ToDouble(DropDownList4.SelectedValue.ToString()),
Convert.ToDouble(TextBox1.Text),
Convert.ToDouble(DropDownList5.SelectedValue.ToString()),
Convert.ToDouble(DropDownList6.SelectedValue.ToString()),
Convert.ToDouble(DropDownList7.SelectedValue.ToString()),
Convert.ToDouble(TextBox2.Text),
Convert.ToDouble(DropDownList8.SelectedValue.ToString()),
Convert.ToDouble(DropDownList14.SelectedValue.ToString()),
Convert.ToDouble(TextBox4.Text),
Convert.ToDouble(DropDownList9.SelectedValue.ToString()),
Convert.ToDouble(TextBox5.Text),
Convert.ToDouble(DropDownList10.SelectedValue.ToString()),
```

Here, the UI page for checking the cancer type calls the classifier functions

This part of the code shows the user interface for patient where patient will input all the necessary details into the form for predicting cancer and this will insert into the database of UserDB.

*Check.aspx.cs × Classifier.cs ×

```
public class Classifier
{
    private DataSet dataSet = new DataSet();

    public DataSet DataSet
    {
        get { return dataSet; }
        set { dataSet = value; }
    }

    public void TrainClassifier(DataTable table)
    {
        dataSet.Tables.Add(table);

        //table
        DataTable GaussianDistribution = dataSet.Tables.Add("Gaussian");
        GaussianDistribution.Columns.Add(table.Columns[0].ColumnName);

        //columns
        for (int i = 1; i < table.Columns.Count; i++)
        {
            GaussianDistribution.Columns.Add(table.Columns[i].ColumnName + "Mean");
            GaussianDistribution.Columns.Add(table.Columns[i].ColumnName + "Variance");
        }

        //calc data
        var results = (from myRow in table.AsEnumerable()
                       group myRow by myRow.Field<string>(table.Columns[0].ColumnName) into g
                       select new { Name = g.Key, Count = g.Count() }).ToList();

        for (int j = 0; j < results.Count; j++)
        {
            DataRow row = GaussianDistribution.Rows.Add();
            row[0] = results[j].Name;

            int a = 1;
            for (int i = 1; i < table.Columns.Count; i++)
            {
                row[a] = Helper.Mean(SelectRows(table, i, string.Format("{0} = '{1}'", table.Columns[0].ColumnName, results[j].Name)));
                row[++a] = Helper.Variance(SelectRows(table, i, string.Format("{0} = '{1}'", table.Columns[0].ColumnName, results[j].Name)));
                a++;
            }
        }
    }

    public string Classify(double[] obj)
    {
        Dictionary<string, double> score = new Dictionary<string, double>();

        var results = (from myRow in dataSet.Tables[0].AsEnumerable()
                       group myRow by myRow.Field<string>(dataSet.Tables[0].Columns[0].ColumnName) into g
```



Here, the classifier calls the Helper function

This part of the code shows how the cancer type will be classified and before classification how the data is being processed and standardized by calling the helper functions.

*Check.aspx.cs × Classifier.cs × Helper.cs ×

```
public static class Helper
{
    public static double Variance(this IEnumerable<double> source)
    {
        double avg = source.Average();
        double d = source.Aggregate(0.0, (total, next) => total += Math.Pow(next - avg, 2));
        return d / (source.Count() - 1);
    }

    public static double Mean(this IEnumerable<double> source)
    {
        if (source.Count() < 1)
            return 0.0;

        double length = source.Count();
        double sum = source.Sum();
        return sum / length;
    }

    public static double NormalDist(double x, double mean, double standard_dev)
    {
        double fact = standard_dev * Math.Sqrt(2.0 * Math.PI);
        double expo = (x - mean) * (x - mean) / (2.0 * standard_dev * standard_dev);
        return Math.Exp(-expo) / fact;
    }

    public static double NORMDIST(double x, double mean, double standard_dev, bool cumulative)
    {
        const double parts = 50000.0; //large enough to make the trapzoids small enough

        double lowBound = 0.0;
        if (cumulative) //do integration: trapezoidal rule used here
        {
            double width = (x - lowBound) / (parts - 1.0);
            double integral = 0.0;
            for (int i = 1; i < parts - 1; i++)
            {
                integral += 0.5 * width * (NormalDist(lowBound + width * i, mean, standard_dev) +
                    (NormalDist(lowBound + width * (i + 1), mean, standard_dev)));
            }
            return integral;
        }
        else //return function value
        {
            return NormalDist(x, mean, standard_dev);
        }
    }
```

Implementation of “Helper” function

This part of the code shows how the Helper function is used to calculate the standard deviation and Mean for cancer data and calculate the Norm Distance to classify the cancer type.

5.3 Interfaces of System Main Functions

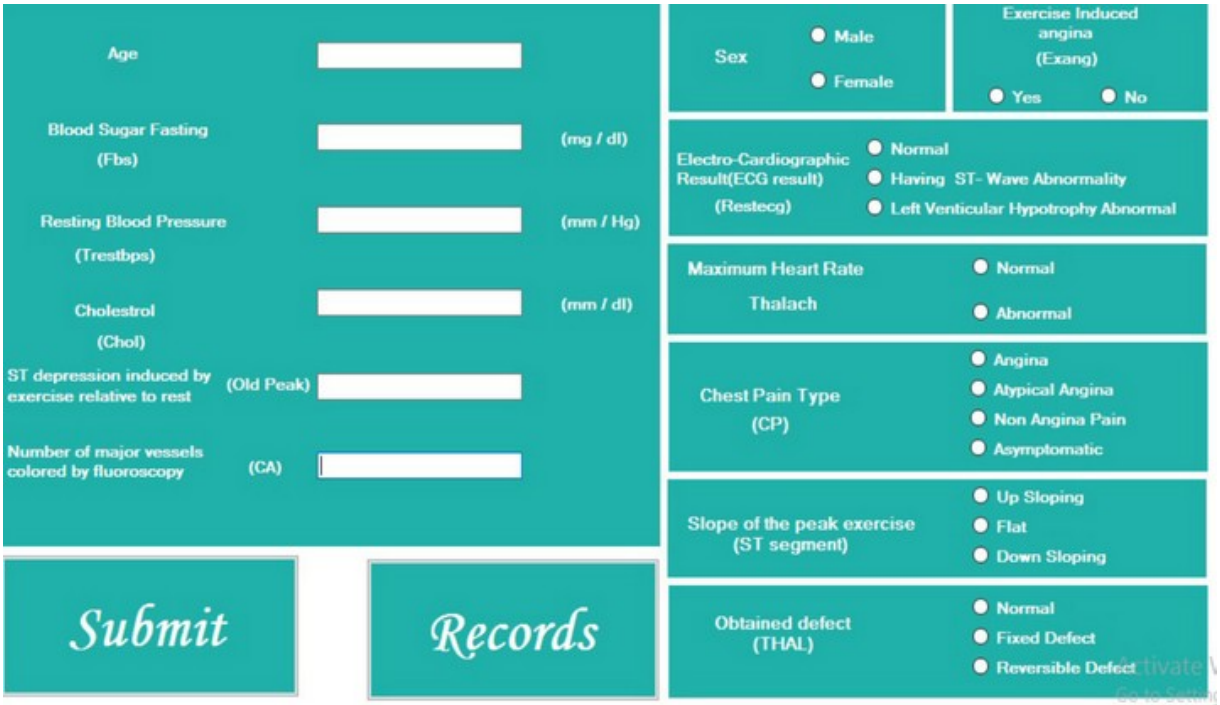


Figure 21 Main system function

Here is the interface of the main system function where it will ask the patients to input all the necessary details. After filling up the details, user will click on the submit button to check the type of cancer that will be predicted.

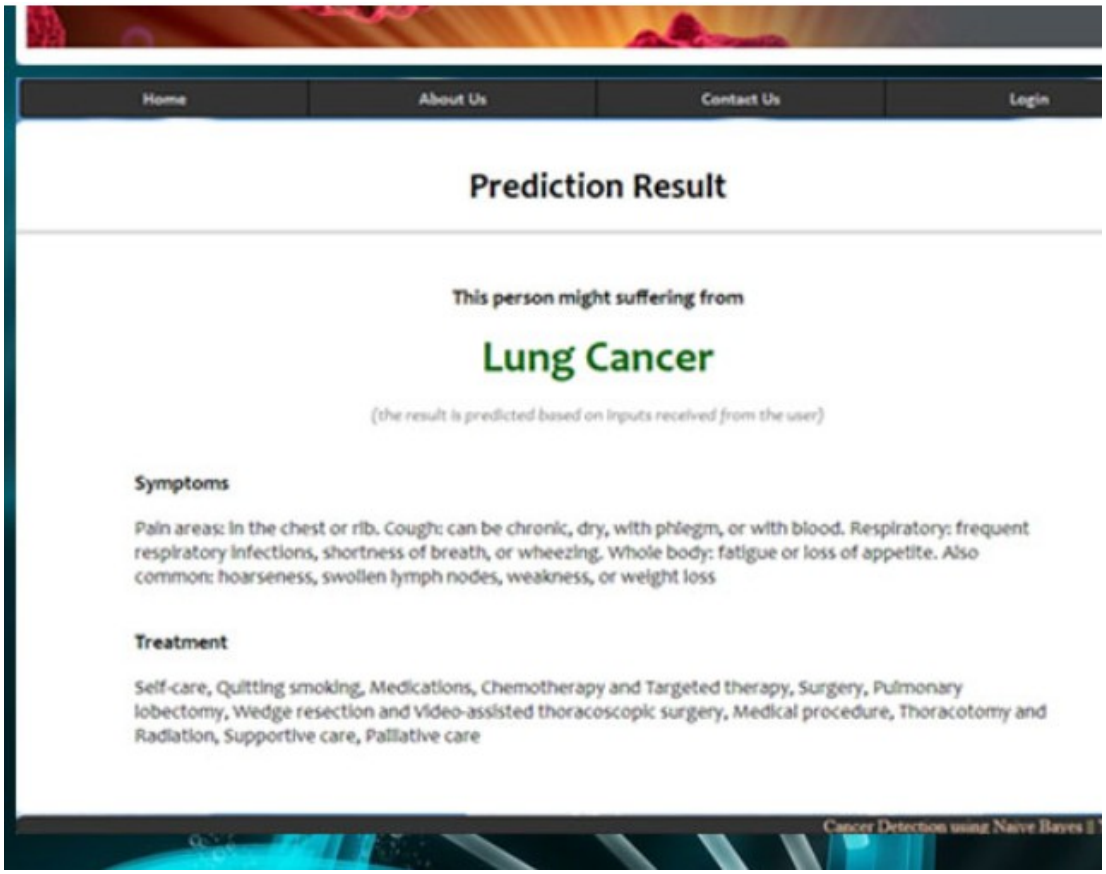


Figure 22 Interface of the result page

After clicking on the submit button, the system will calculate the probability of cancer being predicted depending on the patient's input details and then the system will use data mining techniques and Naive Bayes algorithm to classify the data and predict the cancer type. The system can predict up-to 8 types of cancer.

5.4 Testing

Software testing is used to make sure that this project is bug free and no errors. It involves running software or system components manually or automatically in order to evaluate one or more intriguing characteristics. Software testing will be implemented in this project to find any flaws or unmet requirements compared to the requirements as it has elicited.

For this project software testing will be done using White Box and Black Box Testing. The audience is introduced to software testing in this course, and the significance of software testing is defended.

Testing the module in this project is important because it allows any faults or errors in the application to be found early and fixed before the project is delivered finally. Reliability, security, and high performance are all double confirmed by testing the application, which helps to save time, making it cost efficient, and less hassle for final delivery of this project.

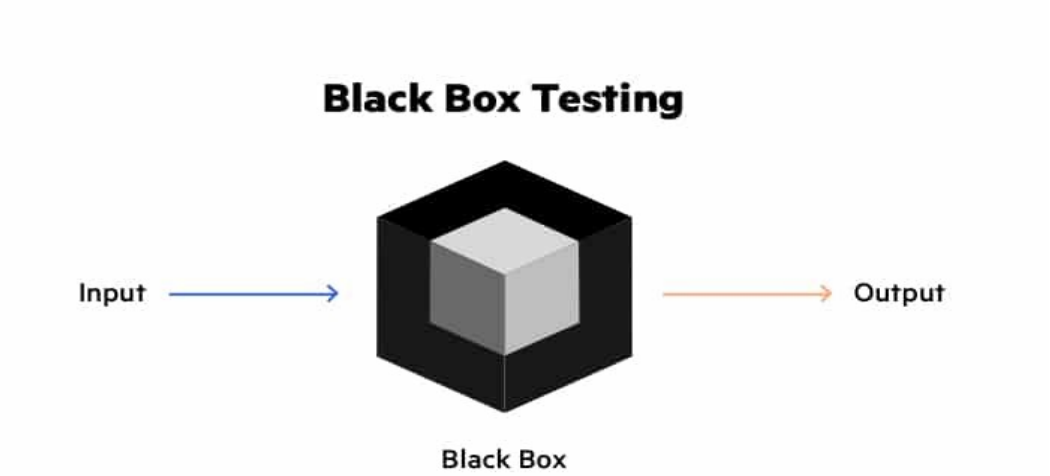
Testing Category	Types of Testing
Functional Testing	• Unit Testing • Integration Testing • Smoke • UAT (User Acceptance Testing) • Localization • Globalization • Interoperability • So on
Non-Functional Testing	• Performance • Endurance • Load • Volume • Scalability • Usability • So on
Maintenance	• Regression • Maintenance

5.4.1 Black box Testing

Black Box testing a testing a system in a "black-box" is doing so without knowing anything about how it operates within. A tester inputs data and monitors the output produced by the system being tested. This allows for the identification of the system's reaction time, usability difficulties, and reliability concerns as well as how the system reacts to anticipated and unexpected user activities.

Because it tests a system from beginning to finish, black box testing is a potent testing method. A tester may imitate user action to check if the system fulfills its promises, much as end users "don't care" how a system is programmed or designed and expect to get a suitable answer to their requests. A black box test assesses every important subsystem along the route, including the UI/UX, database, dependencies, and integrated systems, as well as the web server or application server.

5.4.1.1 System Flow



Black box testing is done for this project to evaluate the system without being aware of how it works inside. This project will be given to admin to explore the entire project under testing to watch the output it produces.

5.4.1.2 Input Output Verification

In order to verify the input and output of the black-box testing, it is important to design the test cases of the project. All the modules' test cases are well-documented in he Software Testing Documentation. They are many ways to carry out the black-box techniques. There are:

I Boundary Value Analysis (BVA)

II Equivalence Partitioning (EP class)

III State Transition Testing

IV Decision Table Testing

For this project, BVA and EP class has been used for the test cases as a testing approach technique in the Software Testing Documentation.

5.4.1.3 Error Messages

There are many error messages that may arise while testing the project. This happens when a developer forget to put the code for handling errors. It often happens becasue when a value is divided by zero, handling with null values, uploading any files without attachment or file uploading when it exceeds the limit size.

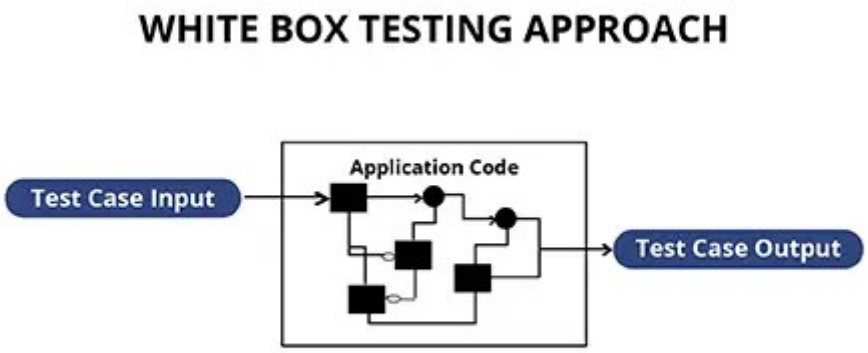
5.4.2 White box Testing

Input-output functionality, design, usability, and security may all be improved by white-box testing, which focuses on examining the program's underlying structure and code. White box testing is also known as clear box testing, open box testing, transparent box testing, code-based testing, and glass box testing since it allows testers to see the source code.

There are two parts to the Box Testing approach to software testing. Black-box testing, on the other hand, is testing from the perspective of the end user. White

box testing in software engineering, on the other hand, focuses on the application's internal workings.

Due to the idea of a see-through box, the name "White-box" was adopted.



5.4.3 User Testing

The success of any system is largely dependent on user acceptability. The system under investigation is put to the test for user approval by regularly communicating with potential system users while it is being developed and making modifications as needed.

5.5 Chapter Summary

This chapter discusses on the main function and objective which is to predicting cancer type. This chapter shows how the implementation of data mining and Naive Bayes algorithm has implemented in this project.

Second, this chapter discussed about two types of testing approaches mainly the White-Box Testing and Black-Box Testing as well as handling errors. Hence, a software testing documentation has been developed with test cases and test approach where the Black-Box testing has been applied with two techniques called the BVA and EP classes.

6

CONCLUSION

6.1 Introduction

Cancer Disease Prediction may be automated by an inadequate internet software application, which has traditionally been seen as a burden in healthcare institutions, wellness centers, and medical offices. Using this technology will assist everyone engaged in scheduling since administrators and users will be able to fulfill their responsibilities more precisely and effectively. The method uses a cancer database to extract hidden information. This system has the potential to be further developed and enhanced.

6.2 Achievement of Project Objectives

The details and objectives of the project has been established successfully throughout the entire semester. Analyzing the project and developing from the existing system has been elicited successfully. The most important milestone of the project was to implement the data mining and classification techniques which has been carried out successfully. The project is able to run all the functional and non-functional requirements successfully which measures and monitors the performance of the project.

The feasibility of the project in terms of technology and expertise is assessed in terms of technical feasibility. The project is completely feasible because it has all the technology and tools that are available to develop the project. As more data will be registering, the system will be expanded more and more and so forth. More data means the system will be more accurate and the feedback option is given in the project so that if there is any glitch in the system, the user will be able to notify the admin.

6.3 Suggestions for Future Improvement

For future improvement, this project needs a huge amount of data which is expensive. Firstly, the system needs to implement in any hospitals or exposing it to any patients in order to get more data so that it can feed the model which can show higher accuracy of predicting the cancer type.

REFERENCES

Ahan Chatterjee, Swagatam Roy, Rupali Shrivastava. "chapter 7 A Machine Learning Approach to Prevent Cancer", IGI Global, 2021

Shanjida Khan Maliha, Romana Rahman Ema, Simanta Kumar Ghosh, Helal Ahmed, Md. Rafsun Jony Mollick, Tajul Islam. "Cancer Disease Prediction Using Naive Bayes, K-Nearest Neighbor and J48 algorithm", 2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT), 2019

Qin, P.. "Dynamic analysis of hydrodynamic bearing-rotor system based on neural network", International Journal of Engineering Science, 200503

Gupta, A. (2015) 'Classification of Complex UCI Datasets Using Machine Learning Algorithms Using Hadoop', *International Journal of Scetific & Techology Research*, 4(5), pp. 85–94.

Huang, C.-L., Chen, Y. H. and Wan, T.-L. J. (2012) 'The mahalanobis taguchi system—adaptive resonance theory neural network algorithm for dynamic product designs', *Journal of Information and Optimization Sciences*, 33(6), pp. 623–635.

Khalid, S., Khalil, T. and Nasreen, S. (2014) 'A survey of feature selection and feature extraction techniques in machine learning', *2014 Science and Information Conference*, pp. 372–378.

Theodoridis, S., Koutroumbas, K., Holmstrom, L. and Koistinen, P. (2009) *Pattern Recognition, Wiley Interdisciplinary Reviews Computational Statistics*.

Zaki, M. J., Wong, L., Berry, M. J. A., Linoff, G. S., Hegland, M., Zaki, M. J. and Wong, L. (2003) 'Data Mining Techniques', *WSPC/Lecture Notes Series: 9in x 6in*, 10(1–2), p. 545.