

CROP MANAGEMENT SYSTEM

Submitted in partial fulfillment of the requirements for the award of
Bachelor of Engineering Degree in Computer Science and Engineering

By

MUKKAMALA VANDANA (37110460)

MUTULURI HARITHA (37110474)



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING SCHOOL OF
COMPUTING**

**SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY JEPPIAAR
NAGAR, RAJIV GANDHI SALAI,
CHENNAI – 600119, TAMILNADU**



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with Grade "A" by NAAC

(Established under Section 3 of UGC Act, 1956)

JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI– 600119

www.sathyabama.ac.in



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Professional training report is the bonafide work of **MUKKAMALA VANDANA (37110460)** and **MUTULURI HARITHA (37110474)** who carried out the project entitled "**CROP MANAGEMENT SYSTEM**" under my supervision from Jan 2021 to April 2021.

Internal Guide

Dr.L.LAKSHMANAN M.E., Ph.D.,

Head of the Department

Dr. S. Vigneshwari, M.E., Ph.D.,

Dr. L. Lakshmanan, M.E., Ph.D.,

Submitted for Viva voce Examination held on _____

Internal Examiner

External Examiner

DECLARATION

I, **MUKKAMALA VANDANA & MUTLURI HARITHA** hereby declare that the Professional training report entitled “ **CROP MANAGEMENT SYSTEM**” done by me under the guidance of **Dr.L.LAKSHMANAN M.E., Ph.D.**, is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in Computer Science and Engineering.

DATE:

PLACE:

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management of SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T.SASIKALA M.E., Ph.D., Dean**, School of Computing , **Dr.S.VIGNESHWARI M.E., Ph.D.,** and **Dr.L.LAKSHMANAN M.E., Ph.D.,** Heads of the Department of Computer Science and Engineering for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere deep sense of gratitude to my project guide **Dr.L.LAKSHMANAN M.E., Ph.D.,**for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Computer Science and Engineering** who were helpful in many ways for the completion of the project.

Certificate of Completion

This is to certify that **Mukkamala Vandana** successfully completed 14.5 total hours of **Learn Python: The Complete Python Programming Course** online course on Aug. 8, 2020

Arinash Jain
Arinash Jain, INSTRUCTOR

&
Udemy

#BeAble

TABLE OF CONTENTS

Title	Page no
	1
INTRODUCTION	
1.1Synopsis	
1.2Objective	
SYSTEM ANALYSIS	
2.1 Existing System	
2.2 Proposed System	
REQUIREMENTS AND SPECIFICATION	
3.1 System Requirement Specification	
3.2 System Requirement	
3.2.1 Hardware Requirements	
3.2.2 Software Requirements	
SOFTWARE DESCRIPTION	
4.1. Features of Dot net	

4.2. The Dot Net framework

4.3. Languages supported by Dot Net

4.4. Objectives of Dot Net Framework

4.5. Features of Dot Net

SYSTEM DESIGN

5.1 Use case diagram

5.2 Activity diagram

5.3 Sequence diagram

5.4 Collaboration diagram

5.5 Data flow diagram

5.6 Architecture diagram

5.7 Entity Relationship diagram

5.8 Class diagram

IMPLEMENTATION

Module description

Home Module

Fertilizer Module

Search fertilizer Module

Crop Module

Crop vs. fertilizer Module

Search crop Module

SYSTEM TESTING

7.1 Testing Objectives

7.2 Types of tests

7.2.1 Unit testing

7.2.2 Integration testing

7.2.3 Functional testing

7.3 System Test

7.4 Test Strategy and Approaches

Conclusion

8.1 Summary

8.2 Future enhancement

ABSTARCT

The crop is basic reason of production of food and raw material, which eventually is reason of survival of the population. In Indian most of the population is dependent on crops. However, there is also need to review and revitalize the mechanism for updating the technology. In the upcoming years agriculture will see major changes. The main purpose for such project is to develop a mobile phone based solution that helps in crop management, leads to agricultural yield improvement and helps in care/maintenance of the crops. The large amount of crop is getting damage in the field due to the bacterial attacks and lack of information resources. Annually, such loss exceeds 40% in total. So, the paper presented here suggest various ways in which a farmer can utilize on their handsets using application called “crop management system”, to assist them for relatively better cultivation and merchandise. Our proposed crop management system application will provides the details about customer and farmer and also it avoids the third party buyer problem which cause problem for farmers. This project used to search For fertilizer and cultivate crop. This helps to update the fertilizer and crop and cultivate. And shows the result of the crop cultivated.

CHAPTER 1

INTRODUCTION

- **ABOUT THE ORGANIZATION**

Wikitechy InfoTech, A Software Company focused in Software Products development, Business Process Outsourcing and Human Resource Consultant with a dictum to provide quality solution to its valuable clients.

We believe in strategic partners rather than solution providers in our professional arena that allows transforming higher business results than fulfilling the business operation alone. Our Service towards the higher vision and dictum provisions the optimal solution to any target with great assurance in this competitive world.

We are open to technological challenges with Strategic scheme that experiences you from a well-planned requirement initiative to quality business deliverables with great efficiency and responsiveness.

WIKITECHY INFOTECH-MISSION

Wikitechy InfoTech an industry leader in the area of information technology. We are a team of highly motivated managers and staff dedicated to providing the best products and services available in the industry. Our Strength is greater from our Commitment towards our clients, our industry, our employee and ourselves.

WIKITECHY INFOTECH-VISION

To Provide Extraordinary Customer Satisfaction through Hassle-free, multi-market project management expertise Adding value at proposal, planning, execution and delivery Advanced technology and tools Superior quality, transparency, responsiveness and accountability

WIKITECHY SOFTWARE SERVICES INCLUDE

WEBSITE MAINTENANCE

Creating a website is easy when compared to maintain it. Over the years we have been developing websites and we have realized that maintaining a site can require more money and effort than developing it. If you don't want to spend your time updating your website, then let us do it for you can focus on your business.

CONTENT DEVELOPMENT AND SEARCH ENGINE OPTIMIZATION

A Website with informative and educational content is a desirable place to visit and will leave your visitors to feel that they have gained something value from it. Content that tells your visitors about your services and products and helps your customers to choose before they buy or select your services.

We have well experienced and ever-growing database of content writers and researchers. Our clients have been excited with the content developed for their websites. Our team of content writers is huge, creative, and professional. They have covered up various topics like hostels, machineries, technologies, resorts, etc.

R&D

Our R&D staff with rich and vivid experience work round the clock with innovative skills, updated software technologies and perfect interpersonal learning, which makes this division an unique one. Our researchers utilize advanced search techniques and carry out research on many search engines and other search tools on the world wide web.

WIKITECHY LIVE PROJECTS

Project Name:

The Canadian Nursing Career Assistance Program. Description: Canadian nursing program is one of our challenging and complicated projects constructed in drupal. Since, customer is enforcing in drupal. We have done the project with the technology specified by the customer. This project focuses towards grouping the entire Nursing community under a single roof. This project incorporates many more functionalities like integrating the existing databases

with our app, social plugin and functional integrations related to Social Networking.

Project Name:

Khalidharr portal Description: A project from Arab Emirates and it's one of a very challenging project and it gave an opportunity for Wikitechy InfoTech to impregnate it's step towards transportation domain. This project is a mix up of PHP API, MySQL, CSS and Java Scripts. This project is in progress with the customer.

Project Name:

Hello Cars Revamp System Description : Hello Cars is one of our Indian projects and it's given to one of the car re-seller company. It's a very common platform to maintain Customer relationship Management, Human resource Management, Data and Information management, financial solution. A comprehensive platform focusing each and every layer of business for the company. This project is primary done in Dotnet 4.0, VS2010 with SQL Server 2008 R2 as back end.

CHAPTER 2

PROBLEM DEFINITION AND FEASIBILITY ANALYSIS

2.1 PROBLEM DEFINITION

Online Crop management system project is aimed to developed an online crop managing information. Every organization has different crop management needs; therefore, we design exclusive crop management systems that are adapted to our managerial requirements. This is designed to assist in strategic planning and will help you ensure that in your organization is equipped with the right level of human

resources for your future goals. This helps the customer to get products to cultivate crop and fertilizer. Which means the allocation of time periods for particular crop and fertilizer is maintained by this application.

2.2 EXISTING SYSTEM

This project helps the farmer to get products for cultivation and also to improve the agriculture. There is no existing project to develop. This gets detailed information to develop for agriculture lands and products.

2.3 PROPOSED SYSTEM

Our proposed crop management system application will provides the details about customer and farmer and also it avoids the third party buyer problem which cause problem for farmers.

2.4 FEASIBILITY ANALYSIS

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

There are three main types of feasibility. They are as follows:

- TECHNICAL FEASIBILITY

- ECONOMIC FEASIBILITY
- OPERATIONAL FEASIBILITY

2.4.1 TECHNICAL FEASIBILITY

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

It measures of the availability of technical resources (hardware components or technical equipment). It also studies the availability of the technical manpower for the project.

Technical feasibility also performs the following tasks.

- Analyzes the technical skills and capabilities of the software development team members

- Determines whether the relevant technology is stable and established. This statement was agreed by our application. The software which is used can able to solve the problem in the existing using the technology.
- Technical skills such as .net languages usage and the connectivity becomes to learn more.
- Backend software Sql server 2008 R2, in this software installation, become more complications such as having the server name, local host number, database name then the connectivity become more tedious before installation.

2.4.2 OPERATIONAL FEASIBILITY

Operational feasibility assesses the extent to which the required software performs a series of steps to solve business problems and user requirements. This feasibility is dependent on human resources (software development team) and involves visualizing whether the software will operate after it is developed and be operative once it is installed. Operational feasibility are generally proposed to have the following properties. They are,

Process: Using this online shipping management application makes the customer to feel friendly.

Evaluation: This process can able to work wherever this application get used.

Implementation: The project gets implemented with calculating the total amount of goods with updated GST tax is displayed and sent to the customer mobile through SMS

Operational feasibility also performs the following tasks.

- Determines whether the problems anticipated in user requirements are of high priority

- Determines whether the solution suggested by the software development team is acceptable
- Analyzes whether users will adapt to a new software
- Determines whether the organization is satisfied by the alternative solutions proposed by the software development team.

Operational Feasibility study is necessary as it ensures that the project developed can be successfully implemented. Everything is provided in easy access manner so that any novice users of information very efficiently.

2.4.3 ECONOMICAL FEASIBILITY

Economic feasibility determines whether the required software is capable of generating financial gains for an organization. It involves the cost incurred on the software development team, estimated cost of hardware and software, cost of performing feasibility study, and so on. For this, it is essential to consider expenses made on purchases (such as hardware purchase) and activities required to carry out software development. In addition, it is necessary to consider the benefits that can be achieved by developing the software. Software is said to be economically feasible if it focuses on the issues listed below.

- Cost incurred on software development to produce long-term gains for an organization
- Cost required to conduct full software investigation (such as requirements elicitation and requirements analysis)
- Cost of hardware, software, development team, and training

Value of product or services

The human needs which are targeted by the project or product are only satisfied during. The last stage of the product or project life cycle, the stage of operation and maintenance.

The services that satisfy the targeted human needs are valued by attaching a market price to them and counting them. In this way, the utility value of the product or service is determined.

Cost of product or services

The various stages in the life cycle of an engineering product or project are also associated with costs, or value sacrificed to create utility value. These costs are incurred during all the stages, namely:

- Planning and design costs
- Manufacture or construction costs
- Operation and maintenance costs.

Accounting profit

The accounting concept of profit is defined as value created in excess of value sacrificed, by an economic unit or investment project, during an accounting period (usually not exceeding one year). This can also be expressed through the accounting principles of revenue (value created during the period), and expense (value sacrificed to create revenue).

2.4.3.1 PLAN OF THE REPORT

Planning plays vital role in any types of project. In this project also planned in such a manner and it has been illustrated as follows.

- 01-12-17 to 15-12-17 Initial study of the problem.
- 16-12-17 to 15-01-18 Assigning the modules for the project.
- 16-01-18 to 19-01-18 Assigning the front end and back end for project.
- 20-01-18 to 21-03-18 Implementation of the project.
- 22-03-18 to 14-04-18 Testing of the project
- 16-04-18 to 31-05-18 Documentation.

CHAPTER 3

SOFTWARE REQUIREMENT SPECIFICATION

3.1 INTRODUCTION

Requirement specification involves studying the current business system to find out how it works and where improvements should be made. The requirement is the feature that must be included in a new system. The analysis of requirements should study and collecting the details of existing system. Software requirements specification (SRS) is the technical way of expressing the requirements for software product. It is treated as the first task of development Process. The proposed system handles additional activities in the existing system.

The software requirements specification is produced at the conclusion of the analysis tasks. The function and performance allocated to software as part of system engineering are refined by establishing a complete information description, detailed functional description, and a representation of system behaviour, on indication of performance requirements and design constraints, appropriate validation criteria and other information relevant to requirements.

3.2 REQUIREMENT ANALYSIS

In system engineering, and software engineering requirements analysis encompasses those tasks that go into determining the needs or conditions to meet for a new or altered product or project, taking account of the possibly conflicting requirements of the various stakeholders, analyzing, documenting, validating and managing software or system requirements.

Requirements analysis is critical to the success or failure of a systems or software project. The requirements should be documented, actionable, measurable, testable, traceable, related to identified business needs or opportunities, and defined to a level of detail sufficient for system design.

3.2.1 PURPOSE

This project is aimed to develop the “Crop Management System” for reducing the manual evaluating the performance of the crop and give the accurate result about crop to manage their periods.

3.2.2 SCOPE

This allows a page to have more links than the out-degree threshold if the cost is reasonable and hence offers a good balance between minimizing changes to a website and reducing information overload to users. Third, we propose two evaluation metrics and use them to assess the improved structure to confirm the validity of our model. The evaluation procedure developed in this paper provides a framework for evaluating website structures win similar studies.

3.2.3 OVERVIEW

The project “**CROP MANAGEMENT SYSTEM**” is used to reduce manual process of the overall system is too clumsy and complicated., time complexity and also

the farmer facing lot of problems to know their schedule details for every periods. It manages every product for the crop stored in database.

3.2.4 GENERAL DESCRIPTIONS

3.2.4.1 PRODUCT FUNCTION

This system is used to manage hours for particular staff's and it can be maintained through online, it consists of many functions i.e. maintain admin module, admin user registration module, login page module, staff detail module, staff vs leave module staff vs subject allocation module, final reallocation module, admin has full authority for managing the entire process.

- **USER CHARACTERISTICS**

The admin has to login into the application by using the login id and password. It displays an admin interface for user authentication. This login page for a project is used to provide a proper genuine authentication to the application with this we can restrict the unauthorized and illegal users in to the application It act as a very great security feature to the application. If the Admin use proper user id and password it enters in to application or else it returns back in to the authentication login page indicating the user is invalid.

- **GENERAL CONSTRAINTS**

- System should have a provision for customer.
- System should have a facility for displaying total amount with updated GST tax for the goods.
- System should allow the admin to maintain the entire process of application.
- System should allow the admin to create report to entire system process.

- **FUNCTIONAL REQUIREMENTS**

The relationship between Input and Output to the system is determined by the functional requirement of the Software Requirement Specifications. The system should provide how the system should react to particular input and how the system should behave in particular situations. Functional requirements may be calculations, technical details, data manipulation and processing and other specific functionality that define what a system is supposed to accomplish.

As in the functional requirements the process involved are accept of submissions in form of customer and perform calculation analysis of the total amount with the GST tax for the goods.

3.2.5.1 TECHNICAL ISSUES

Performance Requirements

Performance depends on two basic factors, they are time complexity and space complexity.

Time

This is one of the most important factors, as the response time has to be minimal to facilitate the user to get an efficiency and speed.

Space

Now a day's storage space has become cheap and is not that much of constraints in software development.

- The performance requirement is specified below:

- The prime requirement is that no error condition causes a project to exit abruptly.
- All the system generated error is directed to a separate page through proper “try and catch block”. The response to any sort of query is fairly fast.
- Any error occurred in any process should return an error messages about the nature of the error that is clear to the user.

- **RISK ANALYSIS**

Risk analysis is a process that helps you identify and manage potential problems that could undermine key business initiatives or projects.

To carry out the risk analysis, you must first identify the possible threats that you face, and then estimate those threats.

Risk analysis can be complex, as you will need to draw on detailed information such as project plans, financial data, security protocols, marketing forecasts and other relevant information. However, it's essential planning tool, and one that could save time, money, and reputations.

- **INTERFACE REQUIREMENTS**

The interface requirement has the purpose of system requirement specification is to produce the specification analysis of the task and also to establish complete information about the requirement, behaviour and other constraints such as functional performance and so on.

During data collection, we investigated and found how the current system operates not only that we also found which are all the problems are faced and how best they can be settled.

The customer described some of the basic requirements of the system that includes calculating the amount and sent through SMS to the customer and viewing report.

3.2.6.1 HARDWARE REQUIREMENTS

- System : Intel inside i3
- System Type : 64-bit Operating System
- Storage : 500GB
- RAM : 4 GB

3.2.6.2 SOFTWARE REQUIREMENTS

- Operating system : Windows 10
- Front End : Microsoft Visual Studio 2010
- IDE : C#.Net
- Data Base : SQL Server 2008 R2

• OTHER FUNCTIONAL ATTRIBUTES

3.2.7.1 SECURITY

Network Applications are available via network access, it is difficult, if not impossible, to limit the population of end-users who may access the application.

In order to protect sensitive content and provide secure modes of data transmission, strong security measures must be implemented.

Security has to be enforced at two levels. At first level, no one other than the person, who knows the user name and password, can be able to logon to the system. At the next level, measures have to be taken so that any normal person does not exercise any of the administrator rights.

3.2.7.2 RELIABILITY

Reliability is the ability of a system to remain operational over time. Reliability is measured as a probability that a system will not fail to perform its intended functions over a specified time interval.

3.2.7.3 MAINTAINABILITY

The term software maintenance is commonly used to refer to the modifications that are made to a software system. This system is developed with ease of maintenance in mind. The modules in this project are clearly documented and have power over the least degree of coupling so that they can be modified independently.

Maintainability is the ease with which a product can be maintained in order to:

- Isolate defects or their cause
- Correct defects or their cause
- Repair or replace faulty or worn-out components without having to replace still working parts
- Prevent unexpected breakdowns
- Maximize a product's useful life

- Maximize efficiency, reliability, and safety
- Meet new requirements
- Make future maintenance easier
- Scope with a changed environment.

3.2.7.4 USABILITY

Usability defines how well the application meet the requirements of the user and consumer by being intuitive, easy to localize and globalize, providing good access for disabled users, and resulting in a good overall user experience.

The application will be developed using standard open source software like java, tomcat web server, internet explorer browser etc. This software will work both on windows and Linux operating system. Hence problems will not arise.

CHAPTER 4

SOFTWARE DESCRIPTION

4.1 FEATURES OF C# .NET

THE .NET FRAMEWORK

The .NET Framework is a new computing platform that simplifies application development in the highly distributed environment of the Internet.

Objectives of .NET FRAMEWORK:

1. To provide a consistent object-oriented programming environment whether object codes is stored and executed locally on Internet-distributed, or executed remotely.

2. To provide a code-execution environment to minimize software deployment and guarantees safe execution of code.
3. Eliminates the performance problems.

There are different types of application, such as Windows-based applications and Web-based applications. To make communication on distributed environment to ensure that code be accessed by the .NET Framework can integrate with any other code.

COMPONENTS OF .NET FRAMEWORK

1. THE COMMON LANGUAGE RUNTIME (CLR):

The common language runtime is the foundation of the .NET Framework. It manages code at execution time, providing important services such as memory management, thread management, and remoting and also ensures more security and robustness. The concept of code management is a fundamental principle of the runtime. Code that targets the runtime is known as managed code, while code that does not target the runtime is known as unmanaged code.

THE .NET FRAMEWORK CLASS LIBRARY:

It is a comprehensive, object-oriented collection of reusable types used to develop applications ranging from traditional command-line or graphical user interface (GUI) applications to applications based on the latest innovations provided by ASP.NET, such as Web Forms and XML Web services.

The .NET Framework can be hosted by unmanaged components that load the common language runtime into their processes and initiate the execution of managed code, thereby creating a software environment that can exploit both managed and

unmanaged features. The .NET Framework not only provides several runtime hosts, but also supports the development of third-party runtime hosts.

Internet Explorer is an example of an unmanaged application that hosts the runtime (in the form of a MIME type extension). Using Internet Explorer to host the runtime to enables embeds managed components or Windows Forms controls in HTML documents.

FEATURES OF THE COMMON LANGUAGE RUNTIME:

The common language runtime manages memory; thread execution, code execution, code safety verification, compilation, and other system services these are all run on CLR.

- Security.
- Robustness.
- Productivity.
- Performance.

Security

The runtime enforces code access security. The security features of the runtime thus enable legitimate Internet-deployed software to be exceptionally feature rich. With regards to security, managed components are awarded varying degrees of trust, depending on a number of factors that include their origin to perform file-access operations, registry-access operations, or other sensitive functions.

ROBUSTNESS:

The runtime also enforces code robustness by implementing a strict type- and code-verification infrastructure called the common type system(CTS). The CTS ensures that

all managed code is self-describing. The managed environment of the runtime eliminates many common software issues.

PRODUCTIVITY:

The runtime also accelerates developer productivity. For example, programmers can write applications in their development language of choice, yet take full advantage of the runtime, the class library, and components written in other languages by other developers.

PERFORMANCE:

The runtime is designed to enhance performance. Although the common language runtime provides many standard runtime services, managed code is never interpreted. A feature called just-in-time (JIT) compiling enables all managed code to run in the native machine language of the system on which it is executing. Finally, the runtime can be hosted by high-performance, server-side applications, such as Microsoft® SQL Server™ and Internet Information Services (IIS).

ASP.NET

ASP.NET is the next version of Active Server Pages (ASP); it is a unified Web development platform that provides the services necessary for developers to build enterprise-class Web applications. While ASP.NET is largely syntax compatible, it also provides a new programming model and infrastructure for more secure, scalable, and stable applications.

ASP.NET is a compiled, .NET-based environment, we can author applications in any .NET compatible language, including Visual Basic .NET, C#, and JScript .NET. Additionally, the entire .NET Framework is available to any ASP.NET application.

Developers can easily access the benefits of these technologies, which include the managed common language runtime environment (CLR), type safety, inheritance, and so on.

ASP.NET has been designed to work seamlessly with WYSIWYG HTML editors and other programming tools, including Microsoft Visual Studio .NET. Not only does this make Web development easier, but it also provides all the benefits that these tools have to offer, including a GUI that developers can use to drop server controls onto a Web page and fully integrated debugging support.

Developers can choose from the following two features when creating an ASP.NET application. Web Forms and Web services or combine these in any way they see fit. Each is supported by the same infrastructure that allows you to use authentication schemes; cache frequently used data, or customizes your application's configuration, to name only a few possibilities.

Web Forms allows us to build powerful forms-based Web pages. When building these pages, we can use ASP.NET server controls to create common UI elements, and program them for common tasks. These controls allow we to rapidly build a Web Form out of reusable built-in or custom components, simplifying the code of a page.

An XML Web service provides the means to access server functionality remotely. Using Web services, businesses can expose programmatic interfaces to their data or business logic, which in turn can be obtained and manipulated by client and server applications. XML Web services enable the exchange of data in client-server or server-server scenarios, using standards like HTTP and XML messaging to move data across firewalls. XML Web services are not tied to a particular component technology or

object-calling convention. As a result, programs written in any language, using any component model, and running on any operating system can access XML Web services

Each of these models can take full advantage of all ASP.NET features, as well as the power of the .NET Framework and .NET Framework common language runtime.

Accessing databases from ASP.NET applications is an often-used technique for displaying data to Web site visitors. ASP.NET makes it easier than ever to access databases for this purpose. It also allows us to manage the database from your code.

ASP.NET provides a simple model that enables Web developers to write logic that runs at the application level. Developers can write this code in the global.aspx text file or in a compiled class deployed as an assembly. This logic can include application-level events, but developers can easily extend this model to suit the needs of their Web application.

ASP.NET provides easy-to-use application and session-state facilities that are familiar to ASP developers and are readily compatible with all other .NET Framework APIs.

ASP.NET offers the IHttpHandler and IHttpModule interfaces. Implementing the IHttpHandler interface gives you a means of interacting with the low-level request and response services of the IIS Web server and provides functionality much like ISAPI extensions, but with a simpler programming model. Implementing the IHttpModule interface allows you to include custom events that participate in every request made to your application.

ASP.NET takes advantage of performance enhancements found in the .NET Framework and common language runtime. Additionally, it has been designed to offer

significant performance improvements over ASP and other Web development platforms. All ASP.NET code is compiled, rather than interpreted, which allows early binding, strong typing, and just-in-time (JIT) compilation to native code, to name only a few of its benefits. ASP.NET is also easily factorable, meaning that developers can remove modules (a session module, for instance) that are not relevant to the application they are developing.

ASP.NET provides extensive caching services (both built-in services and caching APIs). ASP.NET also ships with performance counters that developers and system administrators can monitor to test new applications and gather metrics on existing applications.

Writing custom debug statements to your Web page can help immensely in troubleshooting your application's code. However, it can cause embarrassment if it is not removed. The problem is that removing the debug statements from your pages when your application is ready to be ported to a production server can require significant effort.

ASP.NET offers the Trace Context class, which allows us to write custom debug statements to our pages as we develop them. They appear only when you have enabled tracing for a page or entire application. Enabling tracing also appends details about a request to the page, or, if you so specify, to a custom trace viewer that is stored in the root directory of your application.

The .NET Framework and ASP.NET provide default authorization and authentication schemes for Web applications. we can easily remove, add to, or replace these schemes, depending upon the needs of our application.

ASP.NET configuration settings are stored in XML-based files, which are human readable and writable. Each of our applications can have a distinct configuration file and we can extend the configuration scheme to suit our requirements.

DATA ACCESS WITH ADO.NET

As you develop applications using ADO.NET, you will have different requirements for working with data. You might never need to directly edit an XML file containing data - but it is very useful to understand the data architecture in ADO.NET.

ADO.NET offers several advantages over previous versions of ADO:

- Interoperability
- Maintainability
- Programmability
- Performance Scalability

INTEROPERABILITY:

ADO.NET applications can take advantage of the flexibility and broad acceptance of XML. Because XML is the format for transmitting datasets across the network, any component that can read the XML format can process data. The receiving component need not be an ADO.NET component.

The transmitting component can simply transmit the dataset to its destination without regard to how the receiving component is implemented. The destination component might be a Visual Studio application or any other application implemented with any tool whatsoever.

The only requirement is that the receiving component be able to read XML. SO, XML was designed with exactly this kind of interoperability in mind.

MAINTAINABILITY:

In the life of a deployed system, modest changes are possible, but substantial, Architectural changes are rarely attempted because they are so difficult. As the performance load on a deployed application server grows, system resources can become scarce and response time or throughput can suffer. Faced with this problem, software architects can choose to divide the server's business-logic processing and user-interface processing onto separate tiers on separate machines.

In effect, the application server tier is replaced with two tiers, alleviating the shortage of system resources. If the original application is implemented in ADO.NET using datasets, this transformation is made easier.

ADO.NET data components in Visual Studio encapsulate data access functionality in various ways that help you program more quickly and with fewer mistakes.

PERFORMANCE:

ADO.NET datasets offer performance advantages over ADO disconnected record sets. In ADO.NET data-type conversion is not necessary.

SCALABILITY:

ADO.NET accommodates scalability by encouraging programmers to conserve limited resources. Any ADO.NET application employs disconnected access to data; it does not retain database locks or active database connections for long durations.

VISUAL STUDIO .NET

Visual Studio .NET is a complete set of development tools for building ASP Web applications, XML Web services, desktop applications, and mobile applications. In addition to building high-performing desktop applications, you can use Visual Studio's powerful component-based development tools and other technologies to simplify team-based design, development, and deployment of Enterprise solutions.

Visual Basic .NET, Visual C++ .NET, and Visual C# .NET all use the same integrated development environment (IDE), which allows them to share tools and facilitates in the creation of mixed-language solutions. In addition, these languages leverage the functionality of the .NET Framework and simplify the development of ASP Web applications and XML Web services.

Visual Studio supports the .NET Framework, which provides a common language runtime and unified programming classes; ASP.NET uses these components to create ASP Web applications and XML Web services. Also, it includes MSDN Library, which contains all the documentation for these development tools.

XML WEB SERVICES:

XML Web services are applications that can receive the requested data using XML over HTTP. XML Web services are not tied to a particular component technology or object-calling convention but it can be accessed by any language, component model, or operating system. In Visual Studio .NET, you can quickly create and include XML Web services using Visual Basic, Visual C#, JScript, Managed Extensions for C++, or ATL Server.

XML SUPPORT:

Extensible Markup Language (XML) provides a method for describing structured data. XML is a subset of SGML that is optimized for delivery over the Web. The World Wide

Web Consortium (W3C) defines XML standards so that structured data will be uniform and independent of applications. Visual Studio .NET fully supports XML, providing the XML Designer to make it easier to edit XML and create XML schemas.

VISUAL BASIC .NET

Visual Basic.NET, the latest version of visual basic, includes many new features. The Visual Basic supports interfaces but not implementation inheritance.

Visual basic.net supports implementation inheritance, interfaces and overloading. In addition, Visual Basic .NET supports multithreading concept.

COMMON LANGUAGE SPECIFICATION (CLS):

Visual Basic.NET is also compliant with CLS (Common Language Specification) and supports structured exception handling. CLS is set of rules and constructs that are supported by the CLR (Common Language Runtime). CLR is the runtime environment provided by the .NET Framework; it manages the execution of the code and also makes the development process easier by providing services.

Visual Basic.NET is a CLS-compliant language. Any objects, classes, or components that created in Visual Basic.NET can be used in any other CLS-compliant language. In addition, we can use objects, classes, and components created in other CLS-compliant languages in Visual Basic.NET. The use of CLS ensures complete interoperability among applications, regardless of the languages used to create the application.

IMPLEMENTATION INHERITANCE:

Visual Basic.NET supports implementation inheritance. This means that, while creating applications in Visual Basic.NET, we can drive from another class, which is

known as the base class that the derived class inherits all the methods and properties of the base class. In the derived class, we can either use the existing code of the base class or override the existing code. Therefore, with the help of the implementation inheritance, code can be reused.

CONSTRUCTORS AND DESTRUCTORS:

Constructors are used to initialize objects, whereas destructors are used to destroy them. In other words, destructors are used to release the resources allocated to the object. In Visual Basic.NET the sub finalize procedure is available. The sub finalize procedure is used to complete the tasks that must be performed when an object is destroyed. The sub finalize procedure is called automatically when an object is destroyed. In addition, the sub finalize procedure can be called only from the class it belongs to or from derived classes.

GARBAGE COLLECTION:

Garbage Collection is another new feature in Visual Basic.NET. The .NET Framework monitors allocated resources, such as objects and variables. In addition, the .NET Framework automatically releases memory for reuse by destroying objects that are no longer in use. In Visual Basic.NET, the garbage collector checks for the objects that are not currently in use by applications. When the garbage collector comes across an object that is marked for garbage collection, it releases the memory occupied by the object.

OVERLOADING:

Overloading is another feature in Visual Basic.NET. Overloading enables us to define multiple procedures with the same name, where each procedure has a different

set of arguments. Besides using overloading for procedures, we can use it for constructors and properties in a class.

MULTITHREADING:

Visual Basic.NET also supports multithreading. An application that supports multithreading can handle multiple tasks simultaneously, we can use multithreading to decrease the time taken by an application to respond to user interaction. To decrease the time taken by an application to respond to user interaction, we must ensure that a separate thread in the application handles user interaction.

STRUCTURED EXCEPTION HANDLING:

Visual Basic.NET supports structured handling, which enables us to detect and remove errors at runtime. In Visual Basic.NET, we need to use Try...Catch...Finally statements to create exception handlers. Using Try...Catch...Finally statements, we can create robust and effective exception handlers to improve the performance of our application.

4.2 Features of SQL-SERVER

The OLAP Services feature available in SQL Server version 7.0 is now called SQL Server 2000 Analysis Services. The term OLAP Services has been replaced with the term Analysis Services. Analysis Services also includes a new data mining component. The Repository component available in SQL Server version 7.0 is now called Microsoft SQL Server 2000 Meta Data Services. References to the component now use the term Meta Data Services. The term repository is used only in reference to the repository engine within Meta Data Services

SQL-SERVER database consists of six type of objects,

They are,

1. TABLE
2. QUERY
3. FORM
4. REPORT
5. MACRO

TABLE:

A database is a collection of data about a specific topic.

VIEWS OF TABLE:

We can work with a table in two types,

1. Design View
2. Datasheet View

Design View

To build or modify the structure of a table we work in the table design view. We can specify what kind of data will be hold.

Datasheet View

To add, edit or analyses the data itself we work in tables datasheet view mode.

QUERY:

A query is a question that has to be asked the data. Access gathers data that answers the question from one or more table. The data that make up the answer is

either dynaset (if you edit it) or a snapshot (it cannot be edited). Each time we run query, we get latest information in the dynaset. Access either displays the dynaset or snapshot for us to view or perform an action on it, such as deleting or updating.

FORMS:

A form is used to view and edit information in the database record by record. A form displays only the information we want to see in the way we want to see it. Forms use the familiar controls such as textboxes and checkboxes. This makes viewing and entering data easy.

Views of Form:

We can work with forms in several primarily there are two views,

They are,

1. Design View

2. Form View

Design View

To build or modify the structure of a form, we work in forms design view. We can add control to the form that are bound to fields in a table or query, includes textboxes, option buttons, graphs and pictures.

Form View

The form view which display the whole design of the form.

REPORT:

A report is used to view and print information from the database. The report can group records into many levels and compute totals and average by checking

values from many records at once. Also, the report is attractive and distinctive because we have control over the size and appearance of it.

MACRO:

A macro is a set of actions. Each action in macros does something. Such as opening a form or printing a report. We write macros to automate the common tasks the work easy and save the time.

MODULE:

Modules are units of code written in access basic language. We can write and use module to automate and customize the database in very sophisticated ways.

CHAPTER 5

SYSTEM ANALYSIS

4.1 INTRODUCTION

System analysis is the process of studying a procedure or business in order to identify its goals and purposes and create systems and procedures that will achieve them in an efficient way. Another view sees system analysis as a problem- solving technique that breaks down a system into its component pieces for the purpose of the studying how well those component parts work and interact to accomplish their purpose.

SYSTEM DESIGN

Design is multi-step process that focuses on data structure software architecture, procedural details, (algorithms etc.) and interface between modules.

The design process also translates the requirements into the presentation of software that can be accessed for quality before coding begins.

Computer software design changes continuously as new methods; better analysis and broader understanding evolved. Software Design is at relatively early stage in its revolution.

Therefore, Software Design methodology lacks the depth, flexibility and quantitative nature that are normally associated with more classical engineering disciplines. However, techniques for software designs do exist, criteria for design qualities are available and design notation can be applied.

5.1 INPUT DESIGN

Input design is the process of converting user-originated inputs to a computer-based format. Input design is one of the most expensive phases of the operation of computerized system and is often the major problem of a system.

.

5.2 OUTPUT DESIGN

Output design generally refers to the results and information that are generated by the system for many end-users; output is the main reason for developing the system and the basis on which they evaluate the usefulness of the application. In any system, the output design determines the input to be given to the application.

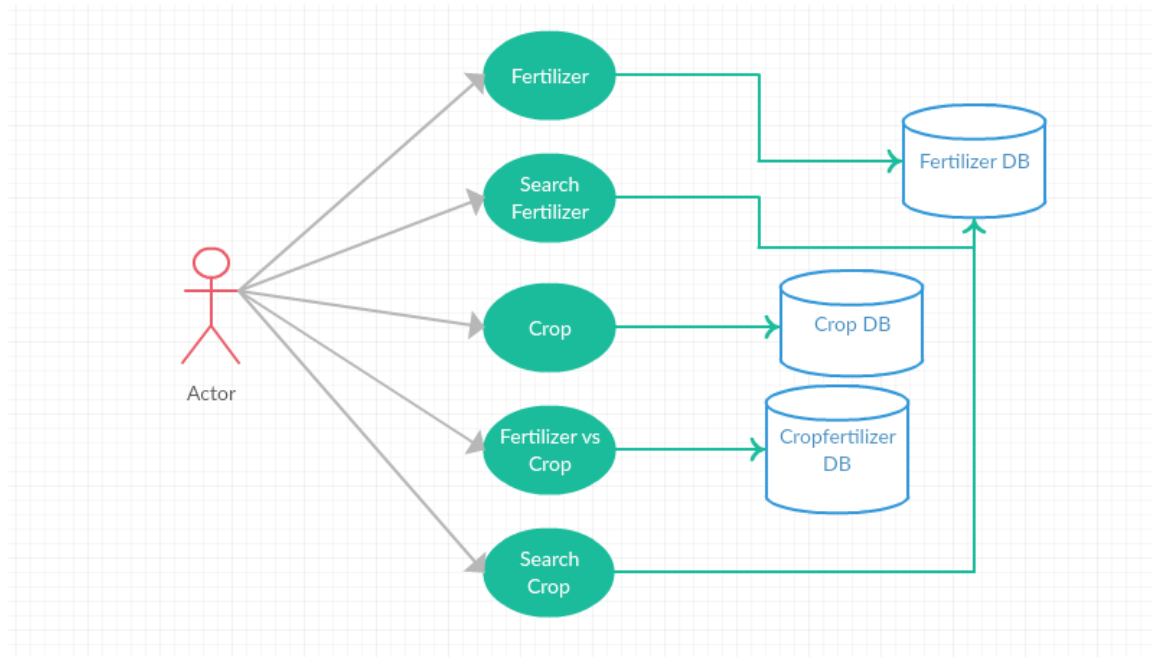
4.2 USECASES AND USE CASE DIAGRAM

A use case diagram at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different use

cases in which the user is involved. A use case diagram can identify the different types of users of a system and the different use cases and will often be accompanied by other types of diagrams as well.

Only static behaviour is not sufficient to model a system rather dynamic behaviour is more important than static behaviour. In UML, there are five diagrams available to model the dynamic nature and use case diagram is one of them. Now as we have to discuss that the use case diagram is dynamic in nature, there should be some internal or external factors for making the interaction.

These internal and external agents are known as actors. Use case diagrams consists of actors, use cases and their relationships. The diagram is used to model the system/subsystem of an application. A single use case diagram captures a particular functionality of a system.



4.3 SEQUENCE DIADRAM

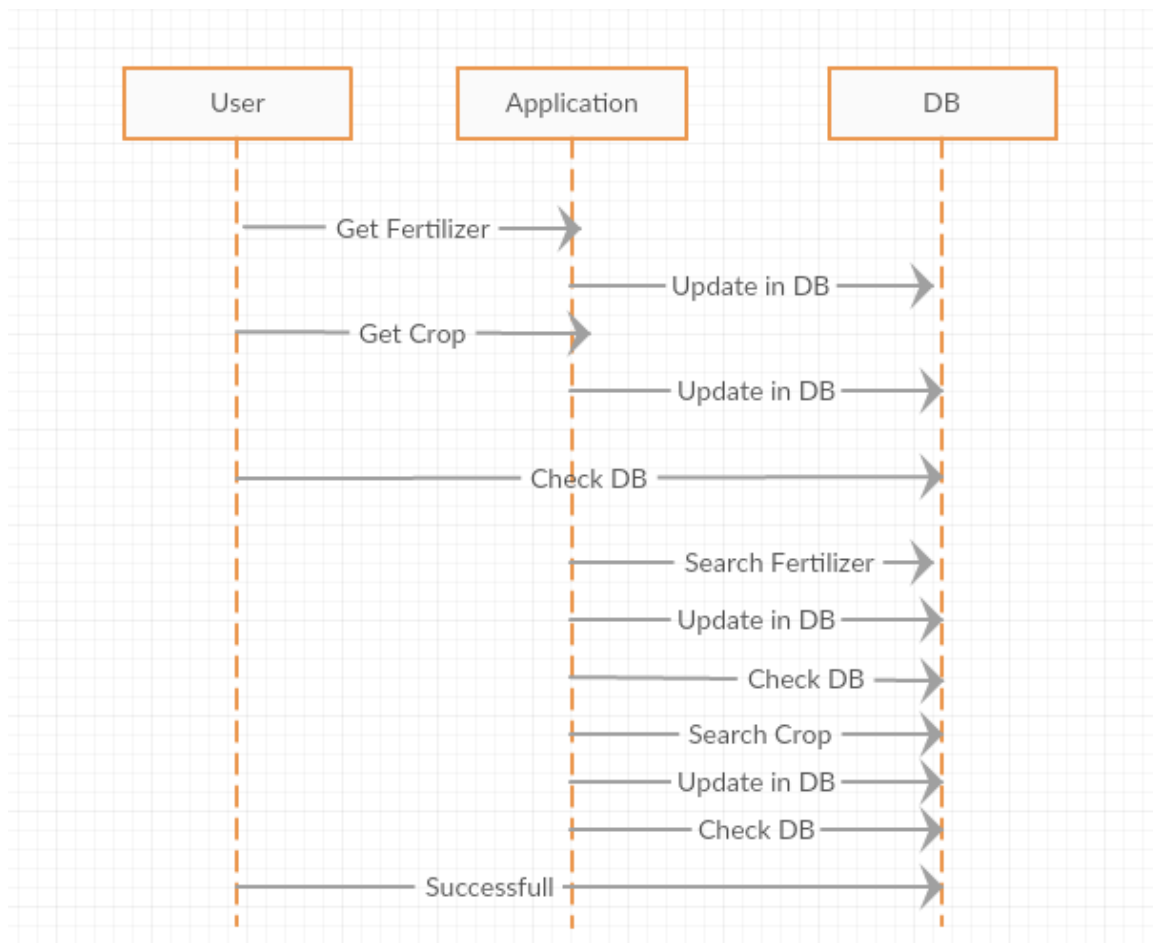
Sequence diagram is the most common kind of interaction diagram, which focuses on the message interchange between a number of lifelines.

Sequence diagram describes an interaction by focusing on the sequence of messages that are exchanged, along with their corresponding occurrence specifications on the lifelines.

UML sequence diagrams are used to show how objects interact in a given situation.

An important characteristic of a sequence diagram is that time passes from top to bottom the interaction starts near the top of the diagram and ends at the bottom (i.e. **Lower equals Later**).

A popular use for them is to document the dynamics in an object-oriented system. For each key collaboration, diagrams are created that show how objects interact in various representative scenarios for that collaboration. The sequence diagram captures the time sequence of the message flow from one object to another.



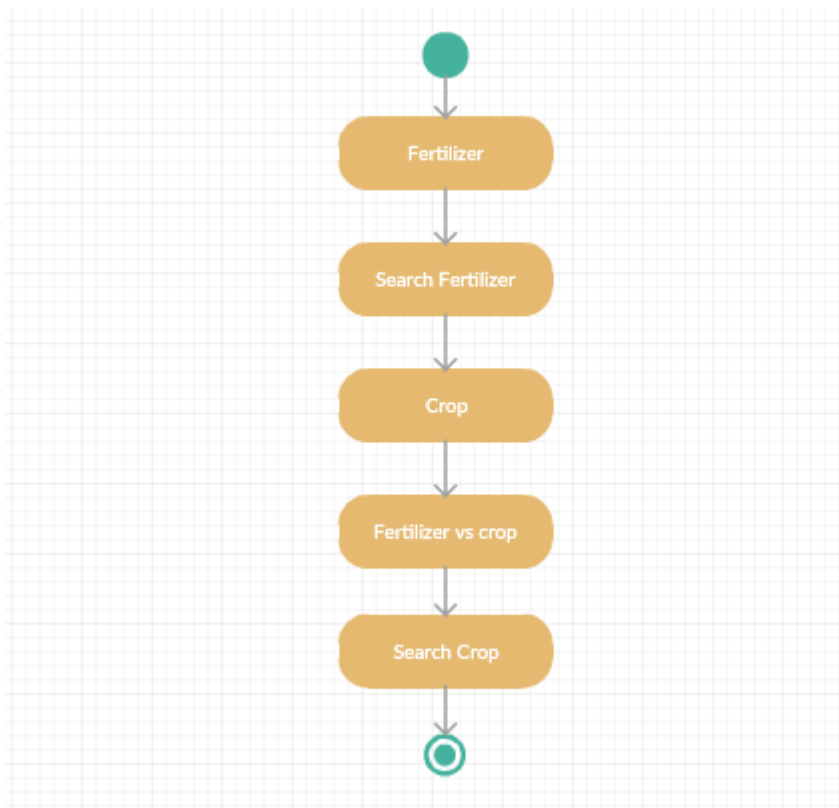
S

4.4 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.

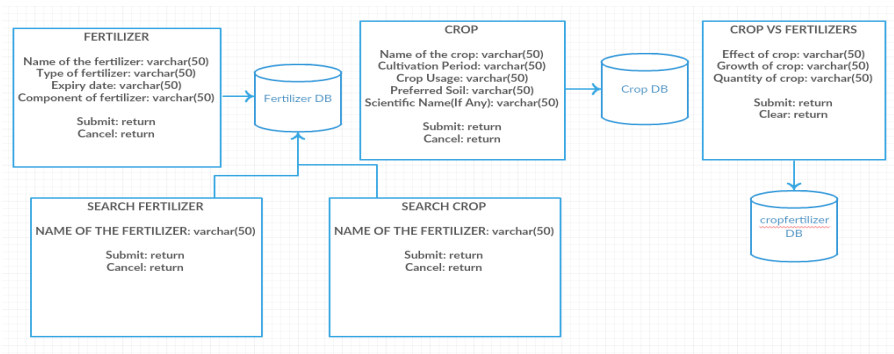
This flow can be sequential, branched, or concurrent. Activity diagrams deal with all type of flow control by using different elements such as fork, join, etc. Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling

Language, activity diagrams are intended to model both computational and organizational processes.



4.5 CLASS DIAGRAM

In software engineering, a class diagram in the UML is a type of static structure diagram that describes the structure of a system by showing the system classes, their attributes, operations and the relationship among the classes. It explains which class contains information's and shows a collection of static model elements such as classes and types, their contents, and their relationship.



CHAPTER 5

SYSTEM DESIGN

5.1 INTRODUCTION

System design is the process of defining the elements of a system such as the architecture, modules and components, the different interfaces of those components and the data that goes through that system. It is meant to satisfy specific needs and requirements of a business or organization through the engineering of a coherent and well-running system.

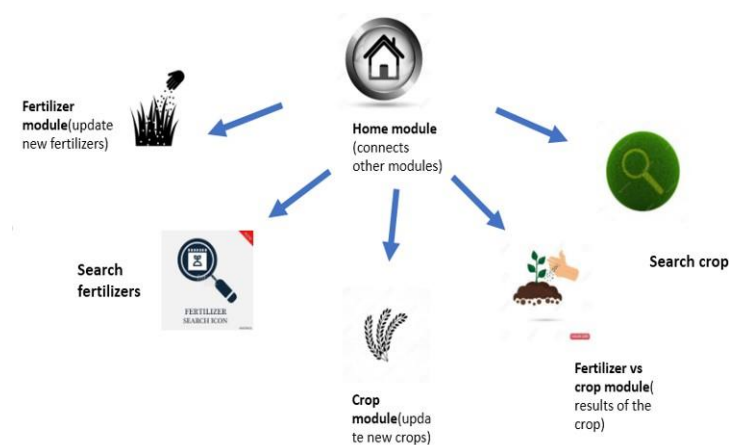
5.2 ARCHITECTURAL DESIGN

The design process for identifying the sub-systems making up a system and the framework for sub-system control and communication is architectural design. The output of this design process is a description of the software architecture.

- It defines an abstraction level at which the designers can specify the functional and performance behaviour of the system.
- It evaluates all top-level designs
- It develops and documents top-level designs for the external and internal interface
- It develops all preliminary versions of users documentation

- It defines and documents preliminary test requirements and the schedule for software integration.

System architecture is a conceptual model that defines the structure, behaviour, and more views of a system. An architecture description is a formal description and representation of a system, organized in a way that supports reasoning about the structure and behaviour of the system.



Architecture Design Representation

Architecture design can be represented using the following models.

1. **Structural model:** Illustrate architecture as an ordered collection of program components.
2. **Dynamic model:** Specifies the behavioural aspect of the software architecture and indicates how the structure or system configuration changes as the function changes due to change in the external environment.
3. **Process model:** Focuses on the design of the business or technical process, which must be implemented in the system.
4. **Functional model:** Represents the functional hierarchy of a system.

5. Framework model: Attempts to identify repeatable architectural design patterns encountered in similar types of application. This leads to an increase in the level of abstraction.

5.3 DATA DESIGN

Data design is the first design activity, which results in less complex, modular and efficient program structure. The information domain model developed during analysis phase is transformed into data structures needed for implementing the software.

The data objects, attributes, and relationships depicted in entity relationship diagrams and the information stored in data dictionary provide a base for data design activity.

During the data design process, data types are specified along with the integrity rules required for the data. For specifying and designing efficient data structures, some principles should be followed. These principles are listed below.

- The data structures needed for implementing the software as well-as the operations that can be applied on them should be identified.
- A data dictionary should be developed to depict how different data objects interact with each other and what constraints are to be imposed on the elements of data structure.
- Stepwise refinement should be used in data design process and detailed design decisions should be made later in the process.
- Only those modules that need to access data stored in a data structure directly should be aware of the representation of the data structure.

- A library containing the set of useful data structures along with the operations that can be performed on them should be maintained.
- Language used for developing the system should support abstract data types.

5.3.1 E-R DIAGRAM

An entity relationship model, also called an entity-relationship (ER) diagram, is a graphical representation of entities and their relationships to each other, typically used in computing in regard to the organization of data within databases or information systems.

5.4 USER INTERFACE DESIGN User interface is the front-end application view to which user interacts in order to use the software. User can manipulate and control the software as well as hardware by means of user interface. Today, user interface is found at almost every place where digital technology exists, right from computers, mobile phones, cars, music players, airplanes, ships etc.

User interface engineering is the design of user interfaces for machines and software, such as computers, home appliances, mobile devices, and other electronic devices, with the focus on maximizing usability and the user experience. The goal of user interface design is to make the user's interaction as simple and efficient as possible, in terms of accomplishing user goals (user-centered design). Interface design is involved in a wide range of projects from computer systems, to cars, to commercial planes; all of these projects involve much of the same basic human interactions yet also require some unique skills and knowledge.

The format of the all interfaces was designed as possible as easy understanding and user friendly. Pages were connected to each other in a consistent

way. Operations that are done with the system were repeatable. The design of the pages allows the users to use this easily.

CHAPTER 6

IMPLEMENTATION

6.1 INTRODUCTION

Implementation is the stage of the project when the theoretical design is turned out into a working system. Thus, it can be considered to be the most critical stage in achieving a successful new system and in giving the user, confidence that the new system will work and be effective.

The implementation stage involves careful planning, investigation of the existing system and its constraints on implementation, designing of methods to achieve changeover and evaluation of changeover methods. Implementation is the most crucial stage in achieving a successful system and giving the user's confidence that the new system is workable and effective. Implementation of a modified application to replace an existing one. This type of conversation is relatively easy to handle, provide there are no major changes in the system.

Each program is tested individually at the time of development using the data and has verified that this program linked together in the way specified in the programs specification, the computer system and its environment is tested to the satisfaction of the user. The system that has been developed is accepted and proved to be satisfactory for the user. And so the system is going to be implemented very soon. A simple operating procedure is included so that the user can understand the different functions clearly and quickly.

Initially as a first step the executable form of the application is to be created and loaded in the common server machine which is accessible to all the user and the server is to be connected to a network. The final stage is to document the entire system which provides components and the operating procedures of the system.

6.2 MODULES SPECIFICATION

A module is a part of a program. Programs are composed of one or more independently developed modules that are not combined until the program is linked. A single module can contain one or several routines. The project contains the following main modules.

- Home Module
- Fertilizer Module
- Search fertilizer Module
- Crop Module
- Crop vs fertilizer Module
- Search crop Module

HOME MODULE

This module the buttons of other modules where connected here and the connected modules are listed below.

- fertilizer module
- search fertilizer module

- crop module
- crop vs fertilizer module
- search module

FERTILIZER MODULE

In fertilizer module, we can update the newly arrived fertilizer for later use then it will store the database of the fertilizers. Fields required in this module are listed below

name of the fertilizer : the fertilizer name

type of fertilizer : the fertilizer type

expiry date : date of expire

components of fertilizer : the components of the fertilizer

SEARCH FERTILIZER MODULE

In search fertilizer Module, we can search the fertilizers using their name which was saved already in the database. Fields required in this module are listed below

name of the fertilizer : fertilizers name

CROP MODULE

The crop module is used to store the Entire details of the crop. Fields Required in this module are listed below:

Name of the crop : Name of the staff

cultivation period : DOB of staff

crop usage : the use of the crop

preferred soil : preferred soil for crop

scientific name(if any) : scientific name of the crop

CROP VS FERTILIZER MODULE

In this module, we can update the results of the crop using the fertilizer. Fields

Required in this module are listed below:

effect of crop : effect of the crop

growth of crop : current growth of the crop

quantity of crop : quantity of fertility

SEARCH CROP MODULE

In search crop Module,we can search the crops using their name which was saved already in the database. Fields required in this module are listed below

name of the crop : crop name

Project Screenshots :





FERTILIZER

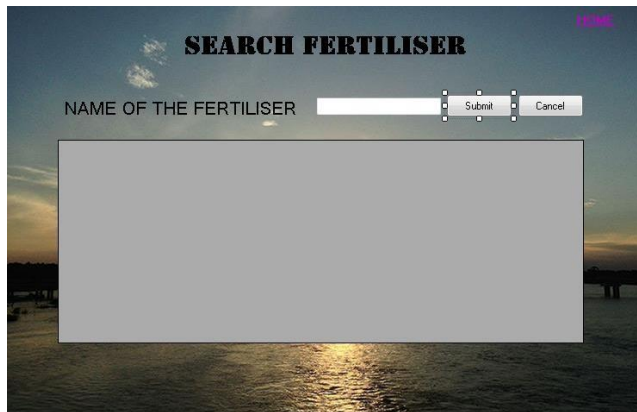
[HOME](#)

Name of the fertilizer

Type of fertilizer

Expiry date

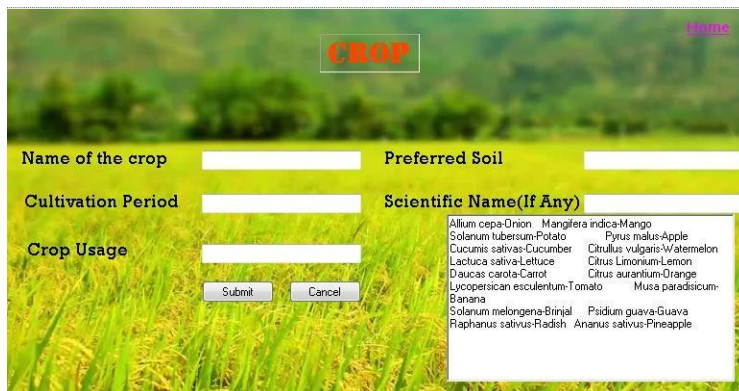
Components of fertilizer



SEARCH FERTILISER

[HOME](#)

NAME OF THE FERTILISER



CROP

[HOME](#)

Name of the crop Preferred Soil

Cultivation Period Scientific Name(If Any)

Crop Usage

Allium cepa-Onion Mangifera indica-Mango

Solanum tuberosum-Potato Pyrus malus-Apple

Cucumis sativus-Cucumber Citrullus vulgaris-Watermelon

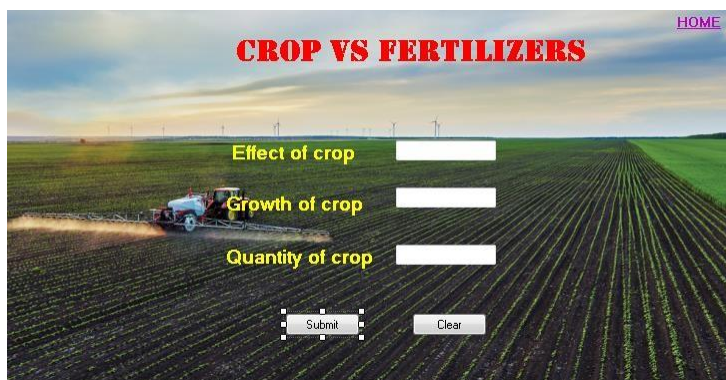
Lactuca sativa-Lettuce Citrus Limonium-Lemon

Daucus carota-Carrot Citrus aurantium-Orange

Lycopersicon esculentum-Tomato Musa paradisiacum-Banana

Solanum melongena-Brinjal Psidium guajava-Guava

Raphanus sativus-Radish Ananas sativus-Pineapple



CROP VS FERTILIZERS

[HOME](#)

Effect of crop

Growth of crop

Quantity of crop

CHAPTER 7

7.1 INTRODUCTION

7.1 TESTING OBJECTIVES

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub assemblies, assemblies and/or a finished product. It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

7.2 TYPES OF TESTS

7.2.1 Unit testing

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application. It is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

7.2.2 Integration testing

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfactory, as shown by successfully unit testing, the combination of components is correct and consistent. Integration

testing is specifically aimed at exposing the problems that arise from the combination of components.

7.2.3 Functional test

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centred on the following items:

Valid Input : identified classes of valid input must be accepted.

Invalid Input : identified classes of invalid input must be rejected.

Functions : identified functions must be exercised.

Output : identified classes of application outputs must be exercised.

Systems/Procedures: interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

Test objectives

- All field entries must work properly.
- Pages must be activated from the identified link.
- The entry screen, messages and responses must not be delayed.

Features to be tested

- Verify that the entries are of the correct format
- No duplicate entries should be allowed
- All links should take the user to the correct page.

7.3 SYSTEM TEST

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An

example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

White Box Testing

White Box Testing is a testing in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is used to test areas that cannot be reached from a black box level.

Black Box Testing

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works.

Unit Testing:

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

7.4 Test strategy and approach

Field testing will be performed manually and functional tests will be written in detail.

Test objectives

- All field entries must work properly.
- Pages must be activated from the identified link.
- The entry screen, messages and responses must not be delayed.

Features to be tested

- Verify that the entries are of the correct format
- No duplicate entries should be allowed

- All links should take the user to the correct page.

Integration Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

Acceptance Testing

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

Test Results:

All the test cases mentioned above passed successfully. No defects encountered.

7.2 TEST PLAN

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies and/or a finished product it is the process of exercising software with the intent of ensuring that the software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

The performance of any system reflects the amount of testing done on it. In every project a crucial part is testing the activities, which help to access and improve

the quality of product that generates during the expansion and modification of the software. Quality attributes include correctness, reliability efficiency and performance of standards and overall cost effectiveness.

Software testing is a critical element of software quality assurance and signifies the final review of specification, design and coding.

7.3 UNIT TESTING

Unit testing involves the design of test cases that validate that the internal that the internal program logic in functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application. It is done after the completion of an individual unit before integration.

This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests performs basic tests at component level and test a specific business process, application, and/or system configuration. Unit test ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

Test strategy and approach. Field testing will be performed manually and functional test will be written in detail.

- Test objectives
- All field entries must work properly

- Pages must be activated from the identified link
- The entry screen, message and responses must not be delayed
- Features to be tested

7.4 INTEGRATION TESTING

The purpose of integration testing is to verify the functional, performance, and reliability between the modules that are integrated. The testing of combined parts of an application to determine if they function correctly together in integration testing.

Integration testing was performed in a regressive way. As and when a module was developed and tested, it was integrated with the remaining modules. Bottom-up testing strategy was followed in each case. Modules were integrated moving upward the control hierarchy.

In “**Crop Management system**” all the modules were tested individually first with testing in all cases. Then they were integrated with the next higher-level module. Within a module, the subsystem was focused on those modules, which were tightly coupled. After a full-fledged integration of all the modules to the main form, the application was tested for interface errors; a correction error etc and was corrected to work successfully.

7.5 SYSTEM TESTING

A system is a regularly interacting or interdependent group of items forming an integrated whole. Every system is delineated by its spatial and temporal boundaries, surrounded and influenced by its environment, described by its structure and purpose and expressed in its functioning.

7.6 VALIIDATION TESTING

The process of evaluating software during the development process or at the end of the development process to determine whether it satisfies specified business requirements.

Validation testing ensures that the product actually meets the clients needs. It can also be defined as to demonstrate that the product fulfils its intended use when deployed on appropriate environment.

Validation testing plays an important role. The application was developed especially for the company and so it is necessary to confirm, whether the right details have been entered? Each and sub-module was tested, whether it is provided the required functionalities? Each input field was tested with the validation rules specified.

Advantages of Validation:

- During verification if some defects are missed then during validation process it can be caught as failures.
- If during verification some specification is misunderstood and development had happened then during validation process while executing the functionality the difference between the actual result and expected result can be understood.

Sample Code:

HOME.CS

```
using System;  
using System.Collections.Generic;  
using System.ComponentModel;  
using System.Data;  
using System.Drawing;  
using System.Linq;
```

```
using System.Text;
using System.Windows.Forms;
namespace fertiliser
{
    public partial class Home : Form
    {
        public Home()
        {
            InitializeComponent();
        }

        private void Form1_Load(object sender, EventArgs e)
        {
        }

        private void label1_Click(object sender, EventArgs e)
        {
        }

        private void button1_Click(object sender, EventArgs e)
        {
            Fertiliser fertile = new Fertiliser ();
            fertile.Show();
            this.Hide();
        }

        private void button2_Click(object sender, EventArgs e)
        {
            Crop crop = new Crop();
            crop.Show();
            this.Hide();
        }

        private void button3_Click(object sender, EventArgs e)
```

```

        {
Cropvsfertilizer crop = newCropvsfertilizer();
crop.Show();
this.Hide();
        }

privatevoid button4_Click(object sender, EventArgs e)
        {
searchfertiliser search = newsearchfertiliser();
search.Show();
this.Hide();
        }
    }
}

```

FERTILIZER.CS

```

using System;
usingSystem.Collections.Generic;
usingSystem.ComponentModel;
usingSystem.Data;
usingSystem.Drawing;
usingSystem.Linq;
usingSystem.Text;
usingSystem.Windows.Forms;
usingSystem.Data.SqlClient;
namespacefertiliser
{
publicpartialclassFertiliser :Form
    {
publicFertiliser()
    {

```

```

InitializeComponent();

    }

private void label3_Click(object sender, EventArgs e)

    {

    }

private void label2_Click(object sender, EventArgs e)

    {

    }

private void textBox4_TextChanged(object sender, EventArgs e)

    {

    }

private void Fertiliser_Load(object sender, EventArgs e)

    {

    }

private void button1_Click(object sender, EventArgs e)

    {

    if (textBox1.Text == "" || textBox2.Text == "" || dateTimePicker1.Text == "" ||
    textBox4.Text == "")

        {

        MessageBox.Show("Please fill the above boxes");

        }

    else

        {

        SqlConnection con = new SqlConnection(@"Data Source=KIT2-PC\VENKAT ;
        Integrated Security = true ; Initial Catalog = fertiliser");

        con.Open();

        SqlCommand cmd = new SqlCommand(@"insert into fertilizer values(" +
        textBox1.Text + "," + textBox2.Text + "," + dateTimePicker1.Text + "," +
        textBox4.Text + ")", con);

        int x = cmd.ExecuteNonQuery();

```

```
if (x > 0)
{
    MessageBox.Show("Data inserted Successfully")
}
else
{
    MessageBox.Show("Data not inserted successfully");
}
con.Close();
}
}

private void linkLabel1_LinkClicked(object sender, LinkLabelLinkClickedEventArgs e)
{
    Home fertiliser = new Home();
    fertiliser.Show();
    this.Hide();
}

private void textBox1_TextChanged(object sender, EventArgs e)
{
}

private void textBox2_TextChanged(object sender, EventArgs e)
{
}

private void dateTimePicker1_ValueChanged(object sender, EventArgs e)
{
}

private void label1_Click(object sender, EventArgs e)
{
}
```

```
    }  
    }  
}
```

SEARCH FERTILIZER.CS

```
using System;  
using System.Collections.Generic;  
using System.ComponentModel;  
using System.Data;  
using System.Drawing;  
using System.Linq;  
using System.Text;  
using System.Windows.Forms;  
using System.Data.SqlClient;  
namespace fertiliser  
{  
    public partial class searchfertiliser : Form  
    {  
        public searchfertiliser()  
        {  
            InitializeComponent();  
        }  
        private void button1_Click(object sender, EventArgs e)  
        {  
            SqlConnection connc = new SqlConnection(@"Data Source =KIT2-PC\VENKAT;  
Integrated Security =true;Initial Catalog=fertiliser");  
            connc.Open();  
            SqlCommand cmd1 = new SqlCommand("Select Name from fertilizer", connc);  
            //SqlDataReader reader = cmd1.ExecuteReader();  
            //AutoCompleteStringCollection MyCollection = new AutoCompleteStringCollection();  
            //while (reader.Read())
```

```

//{
//  MyCollection.Add(reader.GetString(0));
//}

//textBox1.AutoCompleteCustomSource = MyCollection;


string filterval1 = textBox1.Text;

SqlCommand cmd = new SqlCommand(@"select * from fertilizer where Name=" +
filterval1 + "'", connc);

SqlDataAdapter dta = new SqlDataAdapter(cmd);

DataSet dts = new DataSet();

dta.Fill(dts);

        dataGridView1.DataSource = dts.Tables[0];

connc.Close();

    }

private void linkLabel1_LinkClicked(object sender, LinkLabelLinkClickedEventArgs e)
    {
        Home search = new Home();
        search.Show();
        this.Hide();
    }

private void button2_Click(object sender, EventArgs e)
    {
        {
            textBox1.ResetText();
        }
    }

private void dataGridView1_CellContentClick(object sender,
DataGridViewCellEventArgs e)
    {

```

```

    }
private void label2_Click(object sender, EventArgs e)
{
}
}
}

```

CROP.CS

```

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Windows.Forms;

using System.Data.SqlClient;

namespace fertiliser
{
public partial class Crop : Form
{
public Crop()
{
InitializeComponent();
}

private void Crop_Load(object sender, EventArgs e)
{
}

private void label2_Click(object sender, EventArgs e)
{
}
}

```

```
private void linkLabel1_LinkClicked(object sender, LinkLabelLinkClickedEventArgs e)
{
    Home crop = new Home();
    crop.Show();
    this.Hide();
}

private void button1_Click(object sender, EventArgs e)
{
    if (textBox1.Text == "" || textBox2.Text == "" || textBox3.Text == "")
    {
        MessageBox.Show("Please fill the boxes");
    }
    else
    {
        SqlConnection con = new SqlConnection(@"Data Source=KIT2-PC\VENKAT ;
        Integrated Security = true ; Initial Catalog = fertiliser");
        con.Open();

        SqlCommand cmd = new SqlCommand(@"insert into crop values('" + textBox1.Text +
        "','" + textBox2.Text + "','" + textBox3.Text + "')", con);

        int x = cmd.ExecuteNonQuery();

        if (x > 0)
        {
            MessageBox.Show("Data inserted Successfully");
        }
        else
        {
            MessageBox.Show("Data not inserted successfully");
        }
        con.Close();
    }
}
```

```

        }
    }

    private void label1_Click(object sender, EventArgs e)
    {

    }

    private void label3_Click(object sender, EventArgs e)
    {

    }

    private void label4_Click(object sender, EventArgs e)
    {

    }
}

```

CROP VS FERTILIZER.CS

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;

```

```

using System.Text;
using System.Windows.Forms;
using System.Data.SqlClient;

namespace fertiliser
{
    public partial class Cropvsfertilizer : Form
    {
        public Cropvsfertilizer()
        {
            InitializeComponent();
        }

        private void Cropvsfertilizer_Load(object sender, EventArgs e)
        {
        }

        private void label3_Click(object sender, EventArgs e)
        {
        }

        private void label4_Click(object sender, EventArgs e)
        {
        }

        private void button1_Click(object sender, EventArgs e)
        {
            if (textBox1.Text == "" || textBox2.Text == "" || textBox3.Text == "")
            {
                MessageBox.Show("Please fill the details above");
            }
            else
            {

```

```

SqlConnection con = new SqlConnection(@"Data Source=KIT2-PC\VENKAT ;
Integrated Security = true ; Initial Catalog =fertiliser");

con.Open();

SqlCommand cmd = new SqlCommand(@"insert into cropfertiliser values("" +
textBox1.Text + """," + textBox2.Text + """," + textBox3.Text + "")", con);

int x = cmd.ExecuteNonQuery();

if (x > 0)
{
    MessageBox.Show("Data inserted Successfully");
}
else
{
    MessageBox.Show("Data not inserted successfully");
}

con.Close();
}

private void linkLabel1_LinkClicked(object sender, LinkLabelLinkClickedEventArgs e)
{
    Homecropvsfertilizer = new Home();
    cropvsfertilizer.Show();
    this.Hide();
}

private void label1_Click(object sender, EventArgs e)
{
}

private void textBox1_TextChanged(object sender, EventArgs e)
{
}

```

```
private void label2_Click(object sender, EventArgs e)
{

}

private void button2_Click(object sender, EventArgs e)
{
    textBox1.ResetText();
    textBox2.ResetText();
    textBox3.ResetText();
}
}
```

CHAPTER 8

CONCLUSION AND FUTURE ENHANCEMENTS

8.1 CONCLUSION

Technology is introducing new innovations day by day, thus reducing the time required to do things. Shipping is a challenging obstacle for any ecommerce business. The many variables involved in shipping a product from point A to point B means every business shipping strategy will be unique to their product, customers, business and location. The proposed system can be used to reduce the time required to calculate the total amount of goods with currently updated GST tax that is to be imported or exported. The web application provides a way of communication and synchronization between the Customer and the Shipping management system. It also provides them with the facility of communicating with the nearby customer in emergency. The database is a vital aspect of the system. The database of the customer and the shipping system must be checked for consistency on regular basis for smooth working

of the system while the web application for the shipping management system and the customer is also developed using open source tools, hence the system developed is quite feasible.

8.2 FUTURE ENHANCEMENTS

Every application has its own merits and demerits. The project has covered almost all the requirements. Further requirements and improvements can easily be done since the coding is mainly structured or modular in nature. Changing the existing modules or adding new modules can append improvements. Further enhancements can be made to the application, so that the web site functions very attractive and useful manner than the present one.

We enhance the existing system with the social network tagging practice with personalized recommendation model based on combination of user interest and tagging practices This System being web-based application. We decide to develop this web application with the tracking system. To track the ship where the ship is on process to reach the destination. And also We decide to enhance this web application into Android application in future to make easier way to indulge the customer to access our application.

LITERTURE REVIEW

1) TOPIC: survey of text classification algorithms

- In this paper, the author is trying to use generalization and suppression techniques to safeguard the data.
- The data in the system is analyzed for generalization like replacing (or recoding) a value with a less specific but semantically consistent values.
- Suppression is not providing the data to the user.

ADVANTAGES:

- By using generalization and suppression techniques the data can be secured and semantically having consistent values

Drawbacks:

- The major issue with this paper is that, there is no clear explanation on, how the data is going to be secured in suppression technique.
- Considering, the data is not semantically linked. In that case, this technique won't be effective.

2) TOPIC: On Clustering Massive Text and Categorical Data Streams

- Text clustering clearly explains the nature of leakage and possibilities to avoid leakage.
- It specifies how the distributor can “intelligently” give data to agents in order to improve the chances of detecting a guilty agent.
- By adding fake objects to distributed set, the distributor can find the guilty agent easily.
- **ADVANTAGES:**
 - We can get the clear idea to know about the leakage and to detect the guilty agent easily.

Drawbacks:

- This paper didn't explain the kind of impact happening on the system while adding the fake objects.
- Touching or modifying the sensitive data is not an advisable move.

3) TOPIC: A Neighborhood-Based Approach for Clustering of Linked Document Collections

- Clustering can be applied to any database relation having attributes which are such that changes in a few of their values do not affect the applications.

- The tuples attributes within a tuple, bit positions in an attribute, and specific bit values are all algorithmically determined under the control of a private key known only to the owner of the data.

ADVANTAGES:

- The technique used in this paper is flexible and very easy to find the leakage based upon the attributes.

Drawbacks:

- Didn't explain how the knowledge about the schema and Clustering will be given to the other user.
- Not sure how the owner will identify the criticality of the data to be changed.

4) TOPIC: Co-clustering documents and words using Bipartite

- This paper explains about the transformation or modification of data happening automatically due to mining of data or while storing the data in the warehouse.
- Tracing procedures takes advantage of known structure or properties of transformations when present, but also work in the absence of such information.

ADVANTAGES:

- By Using Mining Concepts the Data Modifications Can Be Done While Data Storing and the Structure also defined to trace the transformation.

Drawbacks:

- This paper didn't focus on the latest tools which will solve this kind of problem automatically.
- No clear explain on the security part in this tool.

5) Relational Topic Models for Document Networks

- Document Networks can be applied to any database relation having attributes which are such that changes in a few of their values do not affect the applications.

- The tuples attributes within a tuple, bit positions in an attribute, and specific bit values are all algorithmically determined under the control of a private key known only to the owner of the data.
- **ADVANTAGES:**
- The technique used in this paper is flexible and very easy to find the leakage based upon the attributes.
- **Drawbacks :**
- Didn't explain how the knowledge about the schema and watermark will be given to the other user.
- Not sure how the owner will identify the criticality of the data to be changed.

BIBLIOGRAPHY

BOOK REFERENCE:

- Java Beginner's guide Book by Herbert Schildt
- The Complete Reference Java 2 Fourth Edition Book by Herbert Schildt
- Core Servlets and Java Server Pages Book by Marty Hall
- Web Development with Java Server Pages Book by Duane K.Fields and Mark A.Kolb
- My SQL Database Book by Marc Delisle

WEBSITE REFERENCE:

- Donation page: <https://rcg.org>donate>
- Organ donation: <https://en.m.wikipedia.org>

- Social media and organ donation Research paper, Authors: alexmorgan ,kushalpatel
- www.google.com
- www.w3schools.com
- <http://www.tutorialspoint.com>