INFORMATION SYSTEM FOR VARIETAL EXPERIMENTS

उपजातीय परीक्षणों के लिए सूचना तंत्र

RAKESH KUMAR MESHRAM

INDIAN AGRICULTURAL STATISTICS RESEARCH INSTITUTE
INDIAN AGRICULTURAL RESEARCH INSTITUTE
NEW DELHI – 110012
2009

INFORMATION SYSTEM FOR VARIETAL EXPERIMENTS

BY

RAKESH KUMAR MESHRAM

**Thesis submitted to the Faculty of the Post-Graduate School,
Indian Agricultural Research Institute, New Delhi,
In partial fulfillment of the requirements for the degree of**

**MASTER OF SCIENCE
IN
COMPUTER APPLICATION**

Approved by:

Chairman: ----------
(Dr. V. K. MAHAJAN)

Members: ----------
(Dr. P. K. MALHOTRA)

(Mr. S. B. LAL)

(Mrs. ANU SHARMA)

(Dr. P. K. BATRA)

भारतीय कृषि साँख्यिकी अनुसंधान संस्थान
(भारतीय कृषि अनुसंधान परिषद्) लायब्रेरी एवेन्यू, नई दिल्ली -110012 भारत
INDIAN AGRICULTURAL STATISTICS RESEARCH INSTITUTE
(ICAR)
LIBRARY AVENUE, NEW DELHI-110012 (INDIA)

Dr. V .K. Mahajan
Principal Scientist
Division of Computer Application

CERTIFICATE

This is to certify that the work incorporated in the thesis entitled "INFORMATION SYSTEM FOR VARIETAL EXPERIMENTS (ISVE)", submitted in partial fulfillment of the requirement for the degree of ***Master of Science*** *in* ***Computer Application*** *of the Post Graduate School, Indian Agricultural Research Institute, New Delhi, is a record of bonafide research carried out by* ***Rakesh Kumar Meshram*** *under my guidance and supervision and no part of this dissertation has been submitted for any other degree or diploma.*

All the assistance and help received during the course of this investigation has been duly acknowledged.

Date:

Place: IASRI, New Delhi

(**V. K. Mahajan**)
Chairman
Advisory Committee

ACKNOWLEDGEMENTS

Euripides (484 BC – 406 BC) once told that, "The best and safest thing to keep a balance in your life, acknowledge the great powers around us and in us. If you can do that, and live that way, you are really a wise man." So it is essential that I acknowledge the great powers, which paved the way on which I have walked so far. As a prelude to my thanksgiving, at first I wish to thank the almighty for giving me powers to complete my entire course … after all He is the "greatest".

I am indebted to ***Dr. V. K, Mahajan,*** *Principal Scientist, Division of Computer Application, IASRI, New Delhi, chairman of my advisory committee for his inspiring guidance, support, inquisitiveness, constructive criticisms and constant encouragement throughout the course of this investigation and preparation of the thesis. Without his help, it would not have been possible to complete this work. His very affectionate behavior always filled me with vitality and enthusiasm.*

I am highly grateful to ***Dr. P. K, Malhotra*** *Co-chairman of my Advisory Committee and, Head and Professor (C.A.), IASRI, for providing all the necessary research facilities, valuable suggestions and constant encouragement from time to time for completion of my research work.*

It is great privilege for me to express my esteem and profound sense of gratitude to ***Mr. S. B. Lal Mrs.& Anu Sharma****, Scientist (Sr.Scale.), Division of Computer Applications, and* ***Dr. P. K, Batra,*** *Division of Design of Experiments. for their constructive and valuable suggestions. They were always there in all of my needs and helped their best whenever I seek for it.*

I have great pleasure in expressing my sincere thanks to the valuable guidance and generous help given to me by my advisory committee member,

Thanks are due to ***Dr. V. K, Bhatia****, Director, IASRI, for providing all the facilities throughout the study.*

Words are not enough to thank ***Late Mahendra Khobragade****, friend & philosopher for his help and encouragement. I will always remember him for his love and encouragement.*

I would like to thank my batch mates ***Ramji, Robin, Sanmit Sukanta, Manoj, Mohan Bishal*** *and my seniors* ***Mehrul Alam*** *and* ***Lokesh Dwivedi*** *for supporting my endeavors and for*

encouraging me to do better and better. The words can never express how much I appreciate all that they have done.

I am grateful to my friend ***Radhe and Niraj*** *for their timely and valuable help during my research work.*

I am also indebted to ***Dr. H. S Gupta****, Director, IARI, New Delhi, and* ***Dr. H. S. Gaur****, Dean, PG School, IARI for providing necessary facilities to carry out this work.*

Love, encouragement, and enthusiasm are the fabric stories of intellectual foundation. This is the point, which I have incubated with the blessing of my ***parents****. I bow my mind, body and spirit to be sheltered by the warmest, glorified, golden rays of my beloved parents. I can not express my feeling in words towards my brothers* ***Nilesh, Dinesh*** *and sister* ***Sangeeta*** *and all other family members without those affection and moral support it would not have been possible to reach at this stage of my career. My affectionate thanks are always due to them for their love, affection, support and constant ushering of blessings on me, which were the constant source of inspiration to me.*

Last but not least, I am thankful to ICAR for the financial assistance provided to me in the form of Junior Research Fellowship during the tenure of my study.

Omission, if any in this brief acknowledgement does not imply ingratitude.

Dated:

Place: New Delhi (RAKESH KUMAR MESHRAM)

CONTENTS

CHAPTER I

Introduction

1.1 Background

Agriculture is the most vital sector for the Indian economy contributing to the gross domestic product **(GDP)** of our country. It requires a fairly large database for formulation, implementation and monitoring of research programmes, lab to land programmes, development plans and programmes and management of resources. Computers with their capacity to store large amount of information, ability to retrieve pieces of required information and its processing and dissemination of information on computer networks have opened up vast potentialities for agricultural research development and management. Computers have been used in agriculture and allied fields for over thirty years. Areas of usage advanced from computing to the existing multitude of uses in agricultural research, decision support systems, monitoring, evaluation and control information management and dissemination, teaching and training and many others. The use of computers has brought awareness on the potentials of Computer and Information Technology (IT) in agriculture. The recent advances in computer and communication technology have made computer hardware and software more affordable and user friendly and have resulted in faster movement of information and its utilization. The latest marvel of the modern world is the symbiosis of communication and information technology. It is possible to provide information to close group of people through Internet. Information is a Basic Production Factor apart from Land, Labor and Capital and Information having multiplier effect with other factors. Information systems play an important role for informed decision making for policy and planning, effective resource allocation and its utilization, monitoring of the progress of activities, management of development activities etc.

Information Technology (IT) is defined as the technologies involved in collecting, processing, storing, retrieving, disseminating and implementing data and information using Microelectronics, Optics, Telecommunications and Computers. The recent advances in computer and communication technology have made computer hardware and software more affordable and user-friendly and have resulted in faster movement of information and its utilization. One of the major

benefits of this revolution has been the evolution of Computer Based Information System **(CIS).**

The principle component for the purposeful, orchestrated processing of information are information-system constructs that collect, organize, store, process and display information in all its forms (raw data, interpreted data, knowledge, expertise) and formats (text, video and voice). Computerized information systems have influenced nearly all types of organizations, whether small or large, public or private, national, multinational or others. Information systems have been developed in many areas like Inventory control, Production planning, Engineering, Design, Manufacturing accounting, Customer management etc. These are only some of the areas out of the large domain of areas for which Information System **(IS)** can be developed. An Information System can be developed virtually for any field where information is to be handled and one such field is that of agriculture.

1.2 Varietal Experiments Scenario in India

The India has a number of regional stations in diverse agro-ecological areas. The plant breeders' test/evaluate their varieties for undertaking agriculture research stations and generate voluminous information on the performance and stability of the varieties. Out of this experience the AICRP (All India Coordinated Crop Reseach Projects) was designed to share at different research centers the elite breeding material, conduct identical yield trials to assess the best and stable genotype and their suitability for release as variety. On the strength of this working model in 1965, the All India Coordinated Wheat Improvement Project (AICWIP) was started to accelerate the process of varietal development. The project has active multidisciplinary research centers housed in the various SAUs and these centers get financial support to meet 75% of the annual expenditure from the federal government and the remaining from the state. This cost sharing brought about a cohesive, purpose and result oriented network to develop varieties and at annual meeting all the AICWIP researchers exchange the trial data and results.

Main trust in the coordinated wheat improvement project is to test the wheat varieties being developed by different breeding centers all over India and to identify varieties suitable for release in various part of the country. As varietal trials are most critical component of the programme, though multilocational in nature these trials

are not on cultivator's fields. These are conducted on fields of experimental station with adequate facilities under the direct supervision of breeders. Another feature of the programme is to plan the separate series of trials with separate set of varieties for each of the six agro climatic zones of the country. Within each zone separate series of experiment are planned for different combinations of agronomic conditions like, irrigated, rain fed, timely sown and early sown etc., depending upon the performance of varieties, identification for release is made on the basis of three year data.

1.3 Need of Varietal Information System

Computers with their capacity to store large amount of information, ability to retrieve pieces of required information and its processing and dissemination of information on computer networks have opened up vast potentialities for agricultural development. IASRI has developed an information System related to Agricultural field experiment information system (AFEIS) which aims at systematic maintenance of data of field experiments conducted on various aspects of agricultural technology at a central place and retrieval of information on selective basis as per requirements. It was evolved over years as an outcome of National Index of Agricultural Field Experiments Scheme. The pure varietal trials have been excluded from this database system. Various items included in the database are: objective of the experiment, details about treatments, design, cultural and other practices followed general crop conditions, summary results and/or plot wise observations. Though a large number of experiments are being conducted in National Agricultural Research System but this database contains only the information of those experiments that are collected under this project. Presently the database contains the details of over 25,000 experiments conducted on various crops at different research station of the country. The AFEIS database does not contain the information about varietal experiments. Large amount of data on various varietal experiments is also being generated under various crop improvement projects of ICAR and need to be put in the information system. The results of varietal experiments collected and maintained at a central place in compatible form shall be helpful to planners, researchers and extension workers in knowing the results of experiments conducted in past and approaches followed in this field.

1.4 Objectives

Keeping above in view in present investigation an **(ISVE)** has been developed keeping the following objectives in mind:

- To study the existing information systems on variety trials.
- To design the database for information system on varietal experiments.
- To develop user-friendly web based information system on varietal experiments.
- To test and validate the software developed

1.5 About Information System for Varietal Experiments (ISVE)

ISVE is an information system for varietal experiments, which provides information about varietal experiments conducted on wheat crops at different research stations of the country. .

- Electronic library of all the information related to varietal experiments.
- Data management – Online addition, modification and deletion of information
- Report/queries generation

1.6 Utilities

It can work as a link engine between the researchers, extension personnel and farmers.

The major clients are the extension personnel, farmers, researchers and students who may access information related to varietal experiments.

1.7 Plan of Thesis

This thesis deals with the information on varietal experiments conduct in different field crops which are required by the users such as extension personnel, progressive farmers, students, researchers, and policy makers etc. This thesis is divided into six chapters. Chapter-I of the thesis i.e. the present chapter gives introduction to the problem and objectives of the study. In chapter-II, the review of literature, a brief introduction about ISVE and basic web concepts are presented. In

chapter-III, web concepts and software architecture of the ISVE are dealt in detail. An attempt is made to describe all the technologies used in developing the ISVE with well-commented figures and tables. In chapter-IV, database design & development of ISVE is presented. Here an attempt is made to describe the methodologies used in the development of ISVE. Chapter-V deals with the installation of the software and various operations of the software have been detailed out. In Chapter-VI, future scope of the investigation is described. Finally the thesis is concluded with summary, references and an appendix of the source code.

CHAPTER II

Review of Literature

2.1 General Perspective

The country has made impressive progress in the field of agriculture. The ultimate aim of any breeding programme is to develop varieties superior to the existing ones in yield, diseases resistance and other characteristics. The release of a strain for use as a variety is based on conclusive demonstration of its superiority over the best existing varieties (included as checks in the evaluation trials) in yield or in some other feature of economic importance, e.g., disease resistance, drought tolerance, salt tolerance, etc. For these reasons, the strains are evaluated for their performance, disease resistance, quality etc. extensively in multilocation trials. The multilocation trials are conducted under the concerned All India Coordinated Crop Improvement Projects, which play a key role in testing, identification and release of new varieties. But before the new strains are included in multilocation trials, the breeder should evaluate their performance at experimental station to make sure that the new strain is superior to the existing varieties.

Evaluation of a strain for release as a variety consists of various trials and tests to determine its superiority over the best existing varieties in yield and other agronomic traits.

2.2.1 Station Trial

Station trial is conducted by the breeder who has developed new strain(s). Such a trial is often referred to as preliminary yield trial and may be conducted for one or more years. The objective of station trial is to make sure that the new strains developed by a breeder are superior in performance (at that location) to the best available variety for the region before they are included in the trials of the concerned All India Coordinated Crop Improvement Projects.

2.2.2 Multilocation Trials

These trials are conducted under the respective All India Coordinated Crop Improvement Projects. The objective of these trials is to evaluate the performance of

newly developed strains at several locations distributed over a region. Since the soil and climatic conditions show a large variation from one region of the country to other, the country has been divided into several agro climatic zones; each zone consists of areas having similar agro climatic conditions. The number of zones and the zoning pattern vary considerably from one crop to the other. The pattern of zones for a crop is based on several considerations, including the distribution pattern of the crop and the area under it in the various regions.

2.2.3 Initial Varietal Trial (IVT)

A new variety is first included in an initial varietal trial within the zone in which it was developed. The varieties are tested in IVT for one year only, they are promoted to Advance Varietal Trials (AVT) if their performance is outstanding otherwise they are rejected.

2.2.4 Advance Varietal Trials (AVT)

The varieties that meet preset criteria for yield and disease resistance in the IVT of the previous year are promoted to the Advance Varietal Trials. The AVT for any variety is conducted under the same conditions under which it was evaluated in IVT. AVT is conducted at 20-30 different locations within the zone. AVT may be continued for one or more years. Every year, based on the performance in AVT, varieties poor in yeild are rejected. After evaluation for one year in AVT, the outstanding varieties are included in Agronomic Trials.

2.2.5 Agronomic Trials

A variety showing superior performance in AVT during the first year is included in agronomic trials of the respective All India Coordinated Crop Improvement Projects. The purpose of agronomic trials is to determine the suitable date of sowing and the optimum number of irrigations.

2.2.6 National Initial varietal Trials (NIVT)

The national trials are conducted throughout the country in all the zones. The varieties in national trials consist of one variety from each zone that ranked first in that zone in AVT of the previous year. The national trials serve as IVT for a variety

in zones other than that in which it was developed and tested in AVT. The purpose of national trials is to evaluate outstanding varieties of one zone in the other agro climatic zones to see if they perform well in other zones as well.

2.2.7 Adoptive Research Trials

Adoptive Research Trials are conducted on research stations on farm of state governments. The varieties identified by the workshops of the concerned coordinated project are included in these trials. The data from these trials are considered for release of the identified variety as a new variety.

2.3 Information System

Information is data that have been put into a meaningful and useful context and communicated to a recipient who uses it to make decisions. Information involves the communication and reception of intelligence or knowledge. Information system can be defined as a collection of people, procedures, machines, ideas or activities that generate and process data in a manner that will meet the formal requirements of an organization. Technically, information system is a set of organized procedures that, when executed provides information for decision-making and control of the organization.

2.3.1 Aspects of Information System

The information system of an organization recognizes three aspects

The database, i.e. a complete set of relevant data files must be established. Procedures or programs must be developed to access relevant data from the operation done and from the environment and use these data to update the database. Process must be developed whereby the database can be accessed, analyzed, organized and the resulting information reported to the authorities concerned. While evolving an information system, the attempt should be made to record each piece of data once to utilize it in the same form in every possible way to elicit information for the planning and control purposes of the operation.

To be of any use, a **C**omputer-Based **I**nformation **S**ystem **(CIS)** must function properly, be easy to use and suit the organization for which it has been developed. It helps the people to do their job better and more efficiently.

2.3.2 General Categories of Computer Information System

Data Processing System (DPS): These systems carry out straightforward, voluminous, operational data processing for posting results of day-to-day operations. The products of data processing system are generally used by the administrative staff.

Management Information System (MIS): These systems carry out automatic analysis of data, possibly periodic, on demand, or triggered by certain data in the DPS, to generate information useful in managing the business.

Decision Support System: These systems carry out special analyses, which address strategic management decision problems.

The information system on herbicide use in field crops is an **DPS**. It handles voluminous data on varietal experiments, crop varieties, and their yields etc and provides easy access to the users in a window based menu driven form. Its database can be updated from time to time to reflect changes in recommendations based on latest research in agriculture.

2.4 Previous Work Done

A number of Information Systems have been developed at IASRI and elsewhere. Some of them are mentioned below.

1. Kumar, Sanjay (1995). Development of a Computerized Management Information System for Research Farms. This software deals with management of labors, maintenance of farm machineries, irrigation management and fertilizer & manure management.

2. Ahmad, Md.Zubir (2000). Information System on Apple Crop. This software package covers various aspects of recommended package of practices for apple crop, its processed products, storage and area, production, productivity in major apple growing states and for country.

3. Das, S (2000). Development of Pest Management Information System on Cucurbits. This deals with the IPM of all the major insect-pests, diseases and weeds of cucurbits in all over India.

4. Kumar, B (2000). Development of Pest Management Information System on Sugarcane. This deals with the IPM of all the major insect-pests, diseases and weeds of sugarcane in all over India.

5. Prashant, P (2000). Developed a Pest Management Information System on Tomato. This system deals with the IPM of all the major insect pest, disease and weeds of tomato for entire country India.

6. Gajavelli, S. (2002). Developed a Web-Based Information System for Maize Crop. This software deals with the various aspects of maize crop like recommended packages of practices, plant protection and harvest and post harvest technology.

7. Tripathi, S. (2003). Developed an Integral Library Management System. It provides information about acquisition control, cataloguing process, serial control etc. for a library.

8. Yadav, V.K. (2003). Developed an Information System on Pesticide and their Documentation. It covers information about status of pesticide, residues, decontamination description and metabolism toxicity.

9. Goswami, N. (2004). Developed an Agricultural Research Farm Management Information System for IARI Farm, which provides information on various aspects of management for agricultural research farm.

10. Saurabh, C. (2005). Developed an Intercropping experiments information system which provides information about various intercropping experiments in the field of agricultural science.

10. Kumari, A. (2005). Developed an Information System for Major Fruits (Apple, Banana, Mango) of India. This information system consists of general information, statistics, varieties, planting methods, harvesting and storage details, processing details about these fruit crops.

11. Pal, S. (2005). Developed a Decision Support System for Nutrient Management in Crops. It helps farmers in taking decision regarding nutrient management in crops

and other farmer's queries. This has been developed using ASP.NET and the database part is developed using MS Access.

12. Jha, N. (2007). Developed an Information System on Post-Harvest Management of citrus fruits (KINNOW MANDARIN). This information system provides general information, state wise statistics, harvesting methods, storage practices, packaging, transportation modes, processed products, and processing industry details etc. of citrus fruits (Kinnow Mandarin). This has been developed using ASP.NET and the database part is developed using SQL Server.

13. Jana, C. (2008). Developed an Information System on Herbicide Use in Field Crops, which provides information about herbicide recommendation in field crops for weed control.

CHAPTER III

Web-Concepts and Software Architecture

3.1 Web-Concepts

The World Wide Web started in 1989. It is a network composed of files containing links to related files. These links, called hypertext, would allow a reader to find additional information by simply clicking on any hypertext word or phrase within a document. In 1990, the first text-only browsers were developed. As more and more browsers began to support this new specification, HTML quickly became the de facto standard for publishing documents on the Web.

Today, the World Wide Web is still the fastest growing sector of the Internet. From its beginnings as a simple method for sharing research, the Web has captured the imagination of Internet users and business around the world.

3.1.1 Networks

A network is simply a group of nodes connected in a manner that promotes communication between them. Networked computers are most commonly connected by wire of some sort (often coaxial cable or a twisted pair wire like a phone line); however, machines on a network can be linked by virtually any medium. Some networks communicate via radio waves, infrared, or fiber optic cable. The only absolute requirement of a network is that there must be some way for devices to communicate.

A node by definition is any addressable device connected to a network. A unique address (IP address) is assigned to every node on a network to identify it. Without a unique address, there would be no way to communicate with the device. When a message is packaged and transmitted across the network (this package of information is called a packet) the source and destination addresses are included with the message data.

3.1.2 Protocols

Simply put, a protocol is a formal set of rules that must be followed in order to communicate. Many protocols like HTTP, FTP, and TELNET etc exist today, which facilitate the purpose of communication between two nodes.

3.1.2.1 HTTP

The Hypertext Transfer Protocol (HTTP) is a stateless, TCP/IP based protocol used for communicating on the World Wide Web. HTTP defines the precise manner in which Web clients communicate with Web servers. HTTP/1.0 is the most common version in use today.

HTTP Basics

The HTTP protocol follows a very simple request/response paradigm. In short, a conversation between a Web browser and Web server goes something like this: the client opens a connection to the server, the client makes a request to the server, the server responds to the request, and the connection is closed. A simple Web transaction takes place in the following manner.

1. The client opens a connection to the server. The client opens a TCP connection to the server. Since TCP connections are established down in the transport layer of the protocol stack, there is not a lot of HTTP specific activity in this stage. By default, the connection on the server is made to port 80 (the well-known HTTP port), unless otherwise specified.

2. The client makes a request to the server. This is where we get out first look at HTTP syntax. Let's assume that the Web browser makes a very basic request to retrieve an HTML file. The URL entered into the Web browser might look like http://www.awl.com/index.html. However, the HTTP request sent by the browser to the server would look something like this:

GET / index**.**html HTTP / 1.0

This request can be broken into three parts: the request method, the resource name, and the protocol. GET is an HTTP method requesting the server to send a file. /index.html is a relative path to the file being requested. Because a TCP connection to the server was established in stage 1, it is not necessary to fully qualify the file name using http://www.awl.com/index.html. It is assumed that the requested resource resides on the server with which the browser is currently connected. HTTP/1.0 is the name and version of the protocol implemented by the client.

3. The server responds to the request. The server responds with a

HTTP / 1.0 200 OK

Server: Netscape-Enterprise/2.01

Content-Type: text/html

Status code: various header fields

If the file requested in stage 2 is available and the client has proper authorization, the server's response may look.

```
Content-Length: 87
<HTML>
    <HEAD>
    <TITLE> HTTP Request Successful</TITLE>
    </HEAD>
        <BODY>
            It Worked!
        </BODY>
</HTML>
```

The first line of the response indicates the server's protocol and returns a status code stating that the request was fulfilled successfully. The OK message on this line simply provides a brief, human-readable description of the status code 200.

4. The connection is closed. Either the server or the client or both may close the TCP connection. Usually, it is the server that terminates the connection after the response has been sent. Similarly, a browser will often close the connection once the complete response has been received. If the client wishes to make another request, the whole process starts over again.

3.1.3 Connectionless Protocol

HTTP is a connectionless protocol. The difference between a connectionless and a connection-oriented protocol is in the way they handle connections. Using a connectionless protocol, the client opens connection with the server, sends a request, receives a response, and closes the connection. Each request requires its own connection. With a connection-oriented protocol the client connects to the server, sends a request, receives a response, and then holds the connection open in order to service future requests.

3.1.4 Stateless Protocol

A protocol is said to be stateless if it has no memory of prior connections and cannot distinguish one client's request from that of another. In contrast, in a stateful

protocol, the connection is not opened and closed with every request. After the initial login, the server maintains the user's credentials throughout the session. On the other hand, due to its stateless nature, there is no inherent method in HTTP for tracking a client's traversal of a Web site.

3.1.5 TCP/IP

TCP/IP is a collection, or suite, of protocols used to communicate across a network. The entire protocol suite is named TCP/IP after the two original protocols: Transmission Control Protocol (TCP) and Internet Protocol (IP). Typically, the TCP/IP suite is broken down into four layers (Fig 3.1).

Each layer is responsible for a different aspect of the transmission and each layer insulates the layers above it from some detail of network communication. As data moves through the protocol stack, each layer attaches its own information. The information added by each layer is called a header. Once all layers have attached their header to the data, the package of information that results is called a packet.

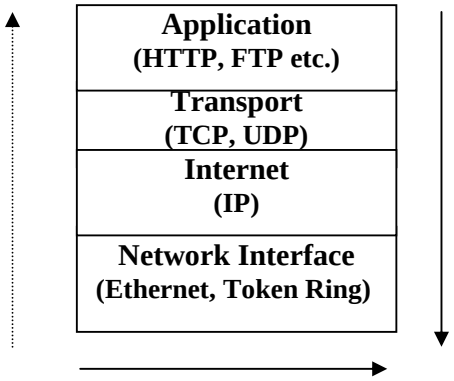

Fig. 3.1 The Four Layers of TCP/IP Model

3.1.6 URLs

Every page on the Internet has a unique Universal Resource Locator (URL) that identifies it. Web browsers request a page by sending a URL to a Web server. The server uses the information in the URL to locate and display the page.

URL syntax is a specific sequence of protocol, domain name, and path to the requested information. The domain name is the Domain Name System (DNS) that identifies the computer containing the information through IP Protocol. The path is

the path to the requested information on the computer. The Table 3.1 shows examples of different URLs:

Table 3.1: Protocol, Domain Name and Path to Information

Protocol	Domain Name	Path to Information
Http://	www.microsoft.com	/back office
Gopher://	gopher.college.edu	/research/astronomy/index.htm
Ftp: //	orion.bureau.gov	/stars/alpha quadrant/starlist.txt

3.2 Three Tier Architecture of ISVE

ISVE is implemented as a layered structure with each layer corresponding to a different functionality. The design of ISVE as given in Figure 3.2 is having following three layers.

- User Interface layer (UIL)
- Application layer (APL)
- Database layer (DBL)

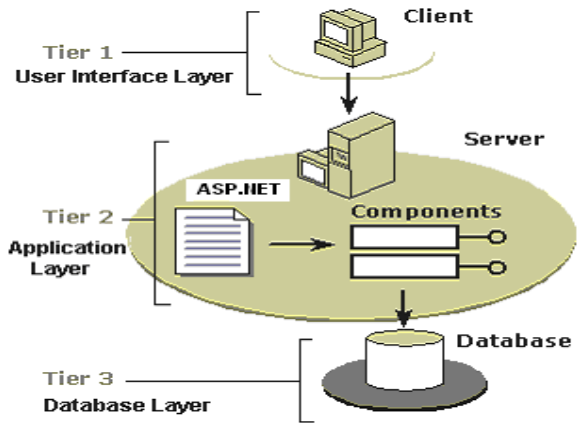

Fig. 3.2 Three Tier Architecture of ISVE

3.2.1 User Interface layer (UIL)

The User Interface Layer is implemented using HTML (Hyper Text Markup Language) and JavaScript. The UIL consists of forms for accepting information from the user and validating those forms using JavaScript.

3.2.1.1 Hyper Text Markup Language (HTML)

HTML or Hypertext Markup Language is the language used to create pages on the World Wide Web (www), known as web pages. Web browsers parse HTML documents and display the contents of the documents based on a set of rules. HTML consists of many tags that describe these rules. The browsers perform actions based on the rules described by the tags. HTML is used to create static web pages. Web browsers, such as Internet Explorer and Netscape, read HTML code and then display text, images, etc. based on the content of the HTML code. For example, the following is a simple line from an HTML document: hello . A browser would parse this line of code and create a paragraph containing the text 'hello'.

For example, to display a bold line saying "hello" in a web page, one would type the following:

Hello

Any text included inside tags should be displayed as bold. Thus, on the web page, the following is displayed:

Hello

Using the tag syntax, which is an opening tag <tag> and an ending tag </tag>, the browser can determine which part of the HTML code upon which it needs to perform a special action. There are four basic tags that should be included in every HTML document. These are the following:

- html
- head
- title
- body

The <html> tag should appear at the beginning and the ending of the document. This tag signifies the beginning and ending of an HTML document. Any information between the <head> and </head> tag is header information. This is where the title of the HTML page should go. The title has its own tag: <title> </title>. The information between the <body> and </body> tag is what gets displayed by the web browser. The following simple HTML template aids in understanding the above:

```
<html>
<head>
   <title>[Enter the title of  page here]</title>
```

```
</head>
<body>
   [Enter the content of page here]
</body>
</html>
```

Tables: HTML tables are one of the most powerful and important features of HTML.

A barebones HTML table has the following format:

```
<table>
   <tr>
      <td>
      </td>
   </tr>
</table>
```

The <table> tags signify the beginning and end of the table. The <tr> tags signify table rows and the <td> tags signify table columns or table cells. This HTML page that makes use of a simple table.

```
<html>
<head>
  <title>Simple table</title>
</head>
<body>
<b>table Example</b>
<table>
   <tr>
      <td>Row 1 Column 1 </td>
      <td>Row 1 Column 2 </td>
   </tr>
   <tr>
      <td>Row 2 Column 1 </td>
      <td>Row 2 Column 2 </td>
   </tr>
</table>
</body>
</html>
```

All in all, tables are a powerful way of displaying text on an HTML page and are a frequently used mechanism of web page authors to perform all kinds of display variants

Links: Link is used for jumping from one web page to another web page. Creating a link is simple. Below is the code used to create the link.

Click here to go to the introduction

The entire code is enclosed inside <a> tags. The "href" attribute of the tag is used to identify the page to which link is to be made. In this case, it is "introduction.html". Once one close beginning <a> tag, the text before the final closing tag will be in the form of a link, which the browser usually displays as underlined text in a color such as blue or red. In the example above, this text was "Click here to go to the introduction".

Apart from these, there are many more tags for making sections and paragraphs, tables, forms, frames, inserting images, character-formatting etc. HTML documents can be broadly classified into static and dynamic documents. Static HTML documents display the same text / images every time while dynamic HTML documents are customized for each user on the fly, based upon their actions or requests. For example, a new employee submits an enrolment form and acknowledgement is dynamically generated that congratulates the employee with the name that he has just typed in the submitted form.

3.2.1.2 JavaScript

JavaScript, created by Netscape, is a scripting language that is commonly used in conjunction with HTML to create dynamic web pages. JavaScript has the unique ability to interact with HTML source code, but has no built-in mechanism for communicating with a server. Thus, JavaScript is most often used as a client-side language to perform such tasks as HTML form validation, while server-side languages like JSP or ASP or ASP.NET are used to facilitate server-side tasks. JavaScript's relatives include VBScript, or Visual Basic Scripting Edition, a language created by Microsoft.

In order to run client-side JavaScript, the code is embedded in the HTML document using <SCRIPT> tag. JavaScript is case sensitive.

Syntax:

```
<SCRIPT LANGUAGE=”JavaScript”>
        {JavaScript Statements}
</SCRIPT>
```

The LANGUAGE attribute is used to specify the scripting language. JavaScript is the default scripting language, so the LANGUAGE definition is optional. JavaScript code starts with the tag <script language="JavaScript"> and ends with the tag </script>.

```
<html>
<head></head>
<body>
<script>
.....// The code embedded in the <body> tags.
</script>
</body>
</html>
```

Another attribute SRC can be used to include an external file containing JavaScript code:

```
<script language="JavaScript" src="hello.js"></script>
```

For example, shown below is the code of the external file hello.js:

```
document.write("Hello World!")
```

The external file is simply a text file containing JavaScript code with the file name extension ".js". Including an external file only function reliably across platforms in the version 4 browsers. The code can't include tags <script language...> and </script>.

3.2.2 Application Layer

Server Side Application Layer is implemented using ASP.NET. ASP.NET is a powerful and flexible technology for creating dynamic Web pages. It is a convergence of two major Microsoft Technologies, Active Server Pages (ASP) and the .NET Framework. Active Server Pages or ASP is a relative old-timer on the Web computing circuit and has provided a sturdy, powerful and effective way of building dynamic Web pages. The .NET Framework, on the other hand, is a whole suite of technologies designed by Microsoft with the aim of revolutionizing the way in which all program development takes place. ASP.NET is a way of creating dynamic Web pages while making use of the innovations present in the .NET Framework.

3.2.2.1 ASP.NET

For years now, Active Server Pages (ASP) has been arguably the leading choice for Web developers building dynamic Websites on Windows Web servers. ASP has gained popularity by offering the simplicity of flexible scripting via several

languages. That combined with the fact that it's built into every Microsoft Windows-based Web server, has made ASP a difficult act to follow.

Early in 2002, Microsoft released its new technology for Internet development. Originally called ASP+, it was finally released as ASP.NET, and represents a leap forward from ASP both in sophistication and productivity for the developer. It continues to offer flexibility in terms of the languages it supports, but instead of a range of simple scripting languages, developers can now choose between several fully-fledged programming languages. Development in ASP.NET requires not only an understanding of HTML and Web design, but also a firm grasp of the concepts of object-oriented programming and development. ASP.NET is a server-side technology for developing Web applications based on the Microsoft .NET.

ASP.NET is server-side; that is, it runs on the Web server. Most Web designing is done by using client-side technologies like HTML, JavaScript, and Cascading Style Sheets (CSS). When a Web browser requests a Web page created with client-side technologies, the Web server simply grabs the files that the browser (the *client*) requests and sends them down the line. The client is entirely responsible for reading the code in the files and interpreting it to display the page on the screen. Server-side technologies, like ASP.NET, are different. Instead of being interpreted by the client, server-side code (for example, the code in an ASP.NET page) is interpreted by the Web server. In the case of ASP.NET, the code in the page is read by the server and used dynamically to generate standard HTML/JavaScript that is then sent to the browser. As all processing of ASP.NET code occurs on the server, it's called a server-side technology. As Figure 3.3 shows, the user (client) only sees the HTML and JavaScript within the browser. The server (and server-side technology) is entirely responsible for processing the dynamic portions of the page.

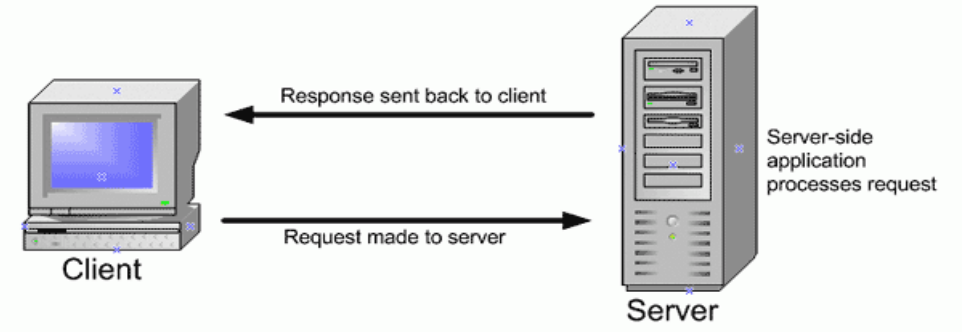

Figure 3.3. Communication between Server and Client

Finally, **ASP.NET is based on the Microsoft .NET Framework.** The .NET Framework collects all the technologies needed for building Windows applications, Web applications, and Web Services into a single package with a set of more than twenty programming languages. To develop Websites with ASP.NET, it is needed to download the .NET Framework Software Development Kit.

3.2.2.2 Inner workings of a typical ASP.NET page

There are some key mechanisms of an ASP.NET page, specifically:

- Page structure
- View state
- Namespaces
- Directives

3.2.2.3 ASP.NET Page Structure

ASP.NET pages are simply text files with the .aspx file name extension that can be placed on an IIS server equipped with ASP.NET. When a browser requests an ASP.NET page, the ASP.NET runtime (as a component of the .NET Framework's Common Language Runtime, or CLR) parses and compiles the target file into a .NET Framework class. The application logic now contained within the new class is used in conjunction with the presentational HTML elements of the ASP.NET page to display dynamic content to the user.

An ASP.NET page consists of the following elements:

- Directives
- Code declaration blocks
- Code render blocks
- ASP.NET server controls
- Server-side comments
- Server-side include directives
- Literal text and HTML tags

It's important to remember that ASP.NET pages are just text files with an .aspx extension that are processed by the runtime to create standard HTML, based on their contents. Presentational elements within the page are contained within the <body>

tag, while application logic or code can be placed inside <script> tags. Figure 3.4 illustrates the various parts of that page.

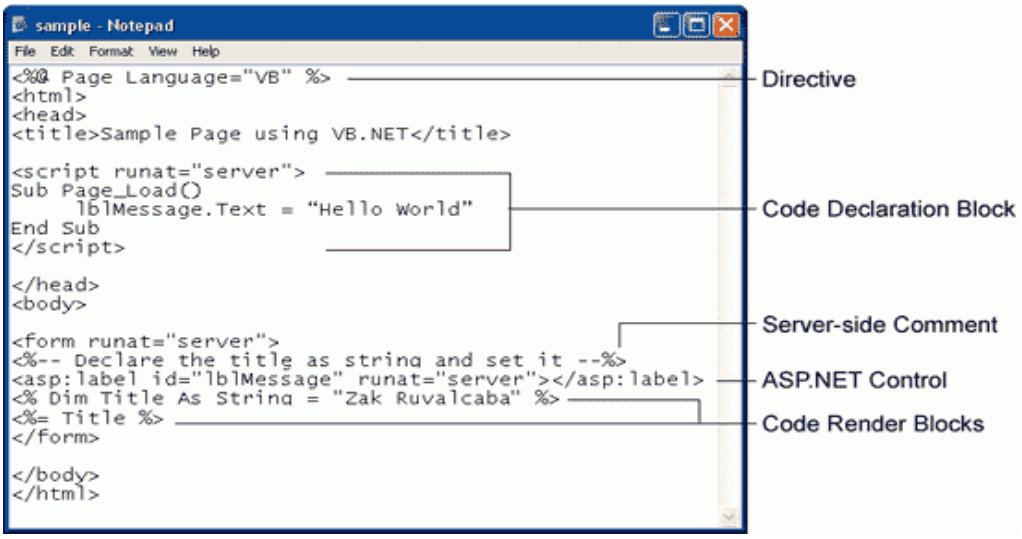

Figure 3.4. An ASP.NET page

This ASP.NET page contains examples of all the above components (except server-side includes) that make up an ASP.NET page.

Directives

The directives section is one of the most important parts of an ASP.NET page. Directives control how a page is compiled, specify settings when navigating between pages, aid in debugging (error-fixing), and allow to import classes to use within page's code. Directives start with the sequence <%@, followed by the directive name, plus any attributes and their corresponding values, then end with %>. Although there are many directives that can be used within pages, the two most important are the Import and Page directives. Import and Page directives are the most useful for ASP.NET development. Looking at the sample ASP.NET page in Figure 3.4, it can be seen that a Page directive was used at the top of the page as shown:

<%@ Page Language="C#" %>

<%@ Page Language=" VB" %>

The Page directive, in this case, specifies the language that's to be used for the application logic by setting the language attribute appropriately. The value

provided for this attribute, in quotes, specifies that laguage used is either VB.NET or C#. There is a whole range of different directives.

Unlike ASP, in ASP.NET, directives can appear anywhere on a page, but are most commonly written as the very first lines.

Code Declaration Blocks

Code declaration blocks must be used to contain all the application logic of ASP.NET page. This application logic defines variables, subroutines, functions, and more. In the sample ASP.NET page, code is placed inside <script> tags:

```
<script runat="server">
Sub mySub()
  ' Code here
End Sub
</script>
```

Here, the tags enclose some VB.NET code, but it could just as easily be C# if our page language were set thus:

```
<script runat="server">
void mySub() {
  // Code here
}
</script>
```

Comments in VB.NET and C# Code

Both of these code snippets contain comments—explanatory text that will be ignored by ASP.NET, but which serves to describe how the code works.

In VB.NET code, a single quote or apostrophe (`'`) indicates that the remainder of the line is to be ignored as a comment.

In C# code, two slashes (`//`) does the same. C# code also lets programer span a comment over multiple lines by beginning it with `/*` and ending it with `*/`.

Code declaration blocks are generally placed inside the <head> tag of ASP.NET page. The sample ASP.NET page shown in Figure 4.2, for instance, contained the following code declaration block:

```
<script runat="server">
```

```
Sub Page_Load()
  lblMessage.Text = "Hello World"
End Sub
</script>
```

The equivalent C# code would be:

```
<script runat="server">
void Page_Load() {
  lblMessage.Text = "Hello World";
}
</script>
```

The <script runat="server"> tag accepts two other attributes, as well. The language used in the block can be set with the language attribute:

```
<script runat="server" language="VB">
<script runat="server" language="C#">
```

If a language is not specified within the code declaration block, the ASP.NET page will use the language provided by the language attribute of the Page directive. Each page may only contain code in a single language; for instance, it is not possible to mix VB.NET and C# in the same page.

The second attribute available is src, which lets programer specify an external code file to use within ASP.NET page:

```
<script runat="server" language="VB" src="mycodefile.vb">
<script runat="server" language="C#" src="mycodefile.cs">
```

Code Render Blocks

Code render blocks are used to define inline code or inline expressions that execute when a page is rendered. Code within a code render block is executed immediately as it is encountered, usually when the page is loaded or rendered for the first time, and every time the page is loaded subsequently. Code within a code declaration block, on the other hand, occurring within script tags, is only executed when it is called or triggered by user or page interactions. There are two types of code render blocks: inline code and inline expressions, both of which are typically written within the body of the ASP.NET page.

Inline code render blocks execute one or more statements and are placed directly inside a page's HTML within <% and %> characters.

Inline expression render blocks can be compared to Response.Write() in classic ASP. They start with <%= and end with %>, and are used to display values of the variables and methods on a page.

Looking back at Figure 3.2, one can see both types of code render blocks:

<% Dim Title As String = "Zak Ruvalcaba" %>

<%= Title %>

This equates to the following C#:

<% String Title = "Zak Ruvalcaba"; %>

<%= Title %>

The first line represents an inline code render block and must contain complete statements in the appropriate language. Here, the value of the Title variable is set to the string Zak Ruvalcaba. The last line is an example of an inline expression render block used to write out the value of the Title variable, Zak Ruvalcaba, onto the page.

ASP.NET Server Controls

At the heart of ASP.NET pages lies the server controls, which represent dynamic elements that users can interact with. There are four basic types of server control: ASP.NET controls, HTML controls, validation controls, and user controls.

All ASP.NET controls must reside within a <form runat="server"> tag in order to function correctly. The only two exceptions to this rule are the HtmlGenericControl and the Label Web control.

Server controls offer the following advantages to ASP.NET developers:

- One can access HTML elements easily from within our code: one can change the characteristics, check the values, or even dynamically update them straight from server-side programming language of choice.
- ASP.NET controls retain their properties even after the page has been processed. This process is known as view state. View state prevents the user from losing data that has already been entered into a form once it’s been sent to the server for processing. When the response comes back to the client’s browser, text box values, drop-down list selections, etc., are all retained through view state.
- With ASP.NET controls, developers are able to separate the presentational elements (everything the user sees) and application logic (dynamic portions of the ASP.NET page) of a page so that each can be considered separately.

Server-Side Comments

Server-side comments allow to include, within the page, comments or notes that will not be processed by ASP.NET. Traditional HTML uses the <!-- and --> character sequences to delimit comments; anything found within these will not be displayed to the user by the browser. ASP.NET comments look very similar, but use the sequences <%-- and -- %>.

The sample ASP.NET example contains the following server-side comment block:

<%-- Declare the title as string and set it --%>

The difference between ASP.NET comments and HTML comments is that ASP.NET comments are not sent to the client at all. In the following example:

```
<!--
<button runat="server" id="myButton" onServerClick="Click">Click
Me</button>
<% Title = "New Title" %>
-->
```

Here, it looks as if a developer has attempted to use an HTML comment to hide not only an HTML button control, but a code render block as well. Unfortunately, HTML comments will only hide things from the browser, not the ASP.NET runtime. So, in this case, while one can not see anything in the browser that represents these two lines, they will, in fact, have been processed by ASP.NET, and the value of the variable Title will be changed to New Title. The code could be modified to use server-side comments very simply:

```
<%--
<button runat="server" id="myButton" onServerClick="Click">Click
Me</button>
<% Title = "New Title" %>
--%>
```

Now, the ASP.NET runtime will ignore the contents of this comment, and the value of the Title variable will not be changed.

Server-Side Include Directives

Server-side includes directives enable developers to insert the contents of an external file anywhere within an ASP.NET page. In the past, developers used server-side

includes when inserting connection strings, constants, and other code that was generally repeated throughout the entire site.

There are two ways that server-side includes can indicate the external file to include: using either the file or the virtual attribute. If file is used, its filename is specified as the physical path on the server, either as an absolute path starting from a drive letter, or as a path relative to the current file. Below, a file server-side include with a relative path is seen:

<! -- #INCLUDE file="myinclude.aspx" -->

virtual server-side, on the other hand, specify the file's location on the Website, either with an absolute path from the root of the site, or with a path relative to the current page. The example below uses an absolute virtual path:

```
<!-- #INCLUDE virtual="/directory1/myinclude.aspx" -->
```

Literal Text and HTML Tags

The final element of an ASP.NET page is plain old text and HTML. Generally, one cannot do without these elements, and HTML is the means for displaying the information from ASP.NET controls and code in a way that is suitable for the user. Returning to the example in Figure 4.2, on focusing on the literal text and HTML tags:

<%@ Page Language="VB" %>
<html>
<head>
<title>Sample Page</title>
<script runat="server">
Sub ShowMessage(s As Object, e As EventArgs)
  lblMessage.Text = "Hello World"
End Sub
</script>
</head>
<body>
<form runat="server">
<%-- Declare the title as string and set it -- %>
<asp: Label id="lblMessage" runat="server" />
<% Dim Title As String = "Zak Ruvalcaba's Book List" %>

```
<%= Title %>
</form>
</body>
</html>
```

As it can be seen in the bold code, literal text and HTML tags provide the structure for presenting dynamic data. Without them, there would be no format to the page, and the browser would be unable to understand it.

View State

ASP.NET controls automatically retain their data when a page is sent to the server by a user clicking a submit button. Microsoft calls this persistence of data *view state*. In the past, developers would have to hack a way to remember the item selected in a drop-down menu or keep the contents of a text box, typically using a hidden form field. This is no longer the case; ASP.NET pages, once submitted to the server for processing, automatically retain all information contained within text boxes, items selected within drop-down menus, radio buttons, and check boxes. Even better, they keep dynamically generated tags, controls, and text. While considering the following ASP.NET page, called sample.aspx:

```
<html>
<head>
  <title>Sample Page using VBScript</title>
</head>
<body>
<form method="post" action="sample.asp">
  <input type="text" name="txtName"/>
  <input type="Submit" name="btnSubmit" text="Click Me"/>
<%
If Request.Form ("txtName") <> "" Then
  Response.Write (Request.Form ("txtName"))
End If
%>
</form>
</body>
</html>
```

If this example is saved in the WebDocs subdirectory of wwwroot it can be opened in browser by typing http://localhost/WebDocs/sample.asp, to see that view state is not automatically preserved. When the user submits the form, the information that was previously typed into the text box is cleared, although it is still available in Request.Form ("txtName"). The equivalent page in ASP.NET, ViewState.aspx, demonstrates data persistence using view state:

Example 3.1. ViewState.aspx

```
<html>
<head>
<title>Sample Page using VB.NET</title>
<script runat="server" language="VB">
Sub Click(s As Object, e As EventArgs)
  lblMessage.Text = txtName.Text
End Sub
</script>
</head>
<body>
<form runat="server">
  <asp: TextBox id="txtName" runat="server" />
  <asp: Button id="btnSubmit" Text="Click Me" OnClick="Click"    runat="server"
/>
  <asp: Label id="lblMessage" runat="server" />
</form>
</body>
</html>
```

Example 3.2. ViewState.aspx

```
<html>
<head>
<title>Sample Page using C#</title>
<script runat="server" language="C#">
void Click(Object s, EventArgs e) {
 lblMessage.Text = txtName.Text; }
</script>
```

```
</head>
<body>
<form runat="server">
  <asp: TextBox id="txtName" runat="server" />
  <asp: Button id="btnSubmit" Text="Click Me" OnClick="Click"    runat="server"
/>
  <asp: Label id="lblMessage" runat="server" />
</form>
</body>
</html>
```

In this case, the code uses ASP.NET controls with the runat="server" attribute. As it can be seen in Figure 3.5, the text from the box appears on the page when the button is clicked, but it can be noticed also that the data remains in the text box. The data in this example is preserved because of view state:

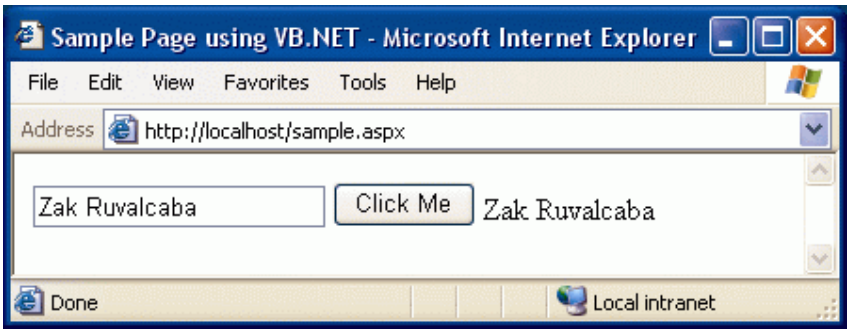

Figure 3.5. View state of ASP.NET page

ASP.NET pages maintain view state by encrypting the data within a hidden form field. The source of the page can be viewed after submission of the form:

```
<input type="hidden" name="__VIEWSTATE" value="dDwtMTcyOTAyO
DAwNzt0PDtsPGk8Mj47PjtsPHQ8O2w8aTwzPjs+O2w8dDxwPGw8aW5uZXJod
G1sOz47bDxIZWxsbyBXb3JsZDs+Pjs7Pjs+Pjs+d2wl7GlhgweO9LlUihS
FaGxk6t4=" />
```

This is a standard HTML hidden form field with the value set to the encrypted data from the form element. As soon as the form is submited for processing, all information relevant to the view state of the page is stored within this hidden form field.

View state is enabled for every page by default. If one do not intend to use view state, it can be turned off, which will result in a slight performance gain in pages. To do this, the EnableViewState property of the Page directive to is to be set false:
<%@ Page EnableViewState="False" %>

Working with Directives

For the most part, ASP.NET pages resemble traditional HTML pages, with a few additions. In essence, just using an extension like .aspx on an HTML file will make the .NET Framework process the page.
Directives control how a page is created, specify settings when navigating between pages, aid in finding errors, and allow to import advanced functionality to use within code. Three of the most commonly used directives are:

Page

Defines page-specific attributes for the ASP.NET page, such as the language used.

Import

Makes functionality defined elsewhere available in a page through the use of namespaces

Register

This directive is used to link a user control to the ASP.NET page.

The Import directive imports extra functionality for use within application logic. The following example, for instance, imports the Mail class, which one could use to send email from a page:

<%@ Import Namespace="System.Web.Mail" %>

The Register directive allows you to register a user control for use on your page. The directive looks something like this:

<%@ Register TagPrefix="uc" TagName="footer" Src="footer.ascx" %>

3.2.3 Database Layer (DBL)

Database Layer is implemented using SQL Server 2000. It is used for designing the following:

- Tables
- Relationships

- Referential Integrity Rules
- Queries

The relational approach is used to design the database. The fundamentals of Normalization Theory are used to normalize the different tables of the database [Loney (2004)]. All tables have proper interaction among themselves via primary key - foreign key relationship.

ADO.NET is the data access technology built into the .NET Framework. Microsoft has created separate namespaces that are optimized for working with different data providers. The following data provider namespaces are included with ADO.NET:

System.Data.SqlClient-	Contains classes for connecting to Microsoft SQL version 7.0 or higher.
System.Data.OleDb-	Contains classes for connecting to a data source that has an OleDb Provider.
System.Data.Odbc –	Contains classes for connecting to a data source that has an ODBC driver.
System.Data.OracleClient-	Contains classes for connecting to an Oracle database server.

In the present IS, SQL Server 2000 database has been used. To connect to a SQL Server database, classes from System.Data.SqlClient namespace are to be used. The System.Data.SqlClient namespace includes the following classes:

SqlConnection- Represents an open database connection to a database.

SqlCommand- Represents a SQL statement or stored procedure.

SqlDataReader- Represents the results from a database query.

3.2.3.1 Performing Common Database Tasks

To import the System.Data.Sqlclient namespace the following page directive is used:

<%@ Import Namespace = “System.Data.SqlClient” %>

3.2.3.2 Opening a Database Connection

To access a database, at first it is needed to create and open a database connection. In Example 3.3, a database connection is created and opened for a SQL sever database named Pubs.

Example : 3.3

```
<%@ Page Language="VB"%>
<%@ Import Namespace="System.Data.SqlClient" %>
<script runat=server>
void Page_Load( object s, EventArgs e )
{
  Dim conpubs as SqlConnection
conpubs=new SqlConnection( Provider=Sqldb value=”Server =(Local);
Trusted_Connection=Yes; initial Catalog=Pubs" )
 conpubs.Open ()
}
</Script>
```

The first imports the necessary namespace, System.Data.Sqlclient, for working with SQL Server. The connection to database is created and opened in the Page_Load () subroutine. First, an instance of the SqlConnection class named conpubs is created. The conpubs class is initialized by passing a connection string as a parameter to the constructor for the SqlConnection class. Finally, the connection is actually opened by calling the Open () method of the SqlConnection class. The connection string contains the name of the provider (Sqldb) for SQL Server and server name, database name etc. to access the database on the server.

3.2.3.3 Retrieving Records from a Database Table

The SQL statement used in ASP.NET pages to retrieve records that match a certain command condition from a database table is Select. Following are the steps to execute a Select statement in an ASP.NET page:

1. Create and open a database connection.
2. Create a database that represents the SQL Select statement to execute.
3. Execute the command with the ExecuteReader () method returning a DataReader.
4. Loop through the DataReader displaying the results of the query.

The ASP.NET page in example 3.4 displays all the records from a SQL database table named pubs.

Example : 3.4

```
<%@ Page Language="VB"%>
<%@ Import Namespace="System.Data.SqlClient" %>
<script runat=server>
Sub Page_Load(Object sender , EventArgs e)
Dim conpubs as SqlConnection
Dim cmdSelect as SqlCommand
Dim reader as SqlDataReader
conpubs = new SqlConnection(Provider=Sqldb value=”Server =(Local);
Trusted_Connection=Yes; initial Catalog=Pubs" )
conpubs.Open()
cmdSelect = new SqlCommand(  "Select au_lname From Authors",  conpubs )
Reader = cmdSelect.ExecuteReader()
Do while  reader.Read()
  Response.Write( "<li>" )
  Response.Write( reader[ "au_lname" ] )
loop
Reader.Close()
conpubs.Close()
End sub
</script>
```

The line <%@ Import Namespace="System.Data.SqlClient" %> imports the necessary namespace to use the ADO.NET classes for the database server. Next, a connection is created and opened for the database located on the local server named Pubs.

After the database connection is opened, a SqlClient object is initialized with a SQL string that contains a SQL Select string. This statement retrieves all the records from a database table named Authors.

Next, the command is executed by calling the ExecuteReader () method of the SqlCommand class. This method returns an SqlDataReader class that represents the results of executing the SQL Select statement.

For an SqlDataReader, it is needed to loop through its contents to display all the records returned by the query. In this example, it is accomplished with a While loop.

All the records returned by the Select statement are displayed with the following block of code:

```
Do while  reader.Read()
  Response.Write( "<li>" )
  Response.Write( reader[ "au_lname" ] )
Loop
```

The Read () method of the SqlDataReader class does two things wherever it is called. First, the method returns the value true if another record exists, but it returns False otherwise. Second, the method advances the DataReader to the next record if a next record exists. By combining these functions, the Read () method enables to quickly loop through the contents of a DataReader. After displaying the records from a DataReader, it should be closed explicitly using the Close () method. Otherwise, the database connection will continue to be tied up.

CHAPTER IV

Database Design

ISVE has been implemented using SQL Server 2000 as been used to design the following:

- Tables
- Queries
- Relationships
- Referential Integrity Rules

The relational approach has been used to design the database. The fundamentals of Normalization Theory have been used to normalize the different tables of the database up to third Normal form. All tables have proper interrelation among themselves via foreign keys. Following database tables along with their fields and size have been used in ISVE.

4.1 Database Tables

Table: Zone

Field Name	Type	Size	Description
Zone_id	Auto number	4	Primary Key
Zone_name	Varchar	50	Name of Zone
Abbreviation	Varchar	10	Abbreviation of zone name

Table: State

Field Name	Type	Size	Description
State_id	Auto number	4	Primary Key
State_name	Varchar	50	Name of State
Zone_id	Int	4	Foreign key of Zone table

Table: Centre

Field Name	Type	Size	Description
Centre_id	Auto number	4	Primary key
Centre_name	Varchar	50	Name of Centre
Zone_id	Int	4	Foreign key of Zone table

Table: Trial Type

Field Name	Type	Size	Description
Trialtype_id	Auto number	4	Primary key
Trialtype_name	Varchar	50	Name of Trial type
Abbreviation	Varchar	10	Abbreviation for Trial type

Table: Trial Series

Field Name	Type	Size	Description
Trialseries_id	Auto number	4	Primary key
Trialseries_name	Varchar	50	Name of Trial type
Abbreviation	Varchar	10	Abbreviation for Trial series

Table: Variety

Field Name	Type	Size	Description
Variety_id	Auto number	4	Primary key
Variety_name	Varchar	50	Name of Trial type
Centre_id	Int	4	Foreign key of Centre table

Table: Var_Tri_Centre

Field Name	Type	Size	Description
Var_tri_centre_id	Auto Number	4	Primary key
Varietal_trial_id	Int	4	Foreign key of Varietal Trial table
Centre_id	Int	4	Foreign key of Centre table
Design_id	Int	4	Foreign key of Design table
Replication	Int	4	Number of Replications
Plot_Size_Gross_length	Int	10	Gross Plot Size (Length)
Plot_Size_Gross_Width	Int	10	Gross Plot Size (Width)
Plot_Size_Net_length	Int	10	Net Plot Size (Length)
Plot_Size_Net_Width	Int	10	Net Plot Size (Width)
Fertlizer_Dose_N	Int	5	Nitrogen
Fertlizer_Dose_P	Int	5	Phosphorus
Fertlizer_Dose_K	Int	5	Potassium
Date_of_sowing	Int	15	Sowing Date
Date_of_Harvesting	Int	15	Harvesting Date
Seed_rate	Int	5	Seed Rate of crop
Seed_requirement	Int	5	Seed Requirement of crop

Table: Design

Field Name	Type	Size	Description
Design_id	Auto number	4	Primary Key
Design_Name	varchar	50	Name of Design

Table: Varietal_Trial

Field Name	Type	Size	Description
Varietal_trial_id	Auto Number	4	Primary key
Trial_series_id	Int	4	Foreign key of Trial Series table
Year	Int	4	Year of Experiment
Zone_id	Int	4	Foreign key of Zone table

Table: Character

Field Name	Type	Size	Description
Character_id	Auto number	4	Primary Key
Character_Name	varchar	50	Name of Character

Table: Var_Trial_Centre_Variety

Field Name	Type	Size	Description
Var_tri_centre_variety_id	Auto Number	4	Primary Key
Var_tri_centre_id	Int	4	Foreign key of Var_Tri_Centre Table
Variety_id	Int	4	Foreign key of Variety Table

Table: Control Variety

Field Name	Type	Size	Description
Control_variety_id	Auto Number	4	Primary Key
Control_variety_name	Int	50	Name of Control Variety
CTL	Text	5	Abbreviation

Table: Var_Tri_CTL

Field Name	Type	Size	Description
Var_tri_CTL_id	Auto Number	4	Primary Key
Control_variety_id	Int	4	Foreign key of Control Variety Table
Varietal_trial_id	Int	4	Foreign key of Varietal Trial Table

Table: AskUs

Field Name	Type	Size	Description
Ask_id	Auto number	4	Primary Key
Email	text	16	E-mail address of user
Query	nvarchar	200	Query of user

Table: FAQ

Field Name	Type	Size	Description
Faq_id	Auto number	4	Primary key
Question	nvarchar	100	Question
Answer	nvarchar	100	Answer

Table: Login

Field Name	Type	Size	Description
First_Name	Varchar	50	First name of user
Last_name	Varchar	50	Last name of user
Login_id	Varchar	15	Primary Key
Password	Varchar	20	Password of user
Designation	varchar	50	Designation of user
Department	Varchar	20	Department of user
Organization	varchar	20	Organization of user

4.2 Queries

The following queries have been used in ISVE for report purpose.

4.2.1 Contributing centers for varieties in different trial series

SELECT dbo.State.State_name, centre.Center_Name, dbo.Variety.Variety_name FROM centre INNER JOIN dbo.Var_trial_centre_variety INNER JOIN dbo.Varietal_trial ON dbo.Var_trial_centre_variety.dbo.Varital_trial_id = dbo.Varietal_trial.Varietal_trial_id INNER JOIN dbo.Variety ON dbo.Var_trial_centre_variety.dbo.Variety_id = dbo.Variety.dbo.Variety_id INNER JOIN dbo.Trial_series ON dbo.Varietal_trial.Trial_series_id = dbo.Trial_series.Trial_series_id ON dbo.centre.Centre_id = dbo.Variety.Centre_id INNER JOIN dbo.State ON dbo.centre.state_id = dbo.State.State_id where dbo.Trial_series.Trial_series_name

4.2.2 Varietal trial conducted during particular year

SELECT dbo.Trial_series. dbo.Trial_series_name AS dbo.Trial_series_name, [Zone]. dbo.Zone_name AS dbo.Zone_name, dbo.Variety.Variety_name AS dbo.Variety_name FROM dbo.Variety INNER JOIN dbo.Var_trial_centre_variety ON dbo.Variety.Variety_id = dbo.Var_trial_centre_variety. dbo.Variety_id INNER JOIN dbo.Varietal_trial INNER JOIN dbo.Trial_series ON dbo.Varietal_trial. dbo.Trial_series_id = dbo.Trial_series. dbo.Trial_series_id INNER JOIN dbo. [Zone] ON dbo.Varietal_trial.zone_id = dbo. [Zone]. dbo.Zone_id ON dbo.Var_trial_centre_variety. dbo.Varital_trial_id = dbo.Varietal_trial. dbo.Varietal_trial_id INNER JOIN dbo.Trial_type ON dbo.Trial_series. dbo.Trial_type_id = dbo.Trial_type. dbo.Trial_type_id WHERE dbo.Trial_type. dbo.Trial_type_name

4.2.3 Experimental Details

SELECT Trial_type.Trial_type_name AS Trial_type_name, Trial_series.Trial_series_name AS Trial_series_name, Varietal_trial.[Year] AS Year, [Zone].Zone_name AS Zone_name, State.State_name AS State_name, Design.Design_name AS Design_name, centre.Center_Name AS Center_Name, Var_trial_centre.Replication_no AS Replication_no,Var_trial_centre.Date_of_sowingfrom AS Date_of_sowingfrom, Var_trial_centre.Date_of_sowingto AS Date_of_sowingto,

```
Var_trial_centre.Plotsize_gross_length AS Plotsize_gross_length,
Var_trial_centre.Seed_requirements as Seed_requirements,
Var_trial_centre.Plotsize_gross_breadth AS Plotsize_gross_breadth,
Var_trial_centre.Plotsize_Net_length AS Plotsize_Net_length,
Var_trial_centre.Plotsize_Net_breadth AS Plotsize_Net_breadth,
Var_trial_centre.Fertilizerdose_N AS Fertilizerdose_N,
Var_trial_centre.Fertilizerdose_K AS Fertilizerdose_K,
Var_trial_centre.Fertilizerdose_P AS Fertilizerdose_P,
Var_trial_centre.Seed_rate AS Seed_rate
FROM        centre INNER JOIN
                  State INNER JOIN
                  Trial_type INNER JOIN
                  Trial_series ON Trial_type.Trial_type_id = Trial_series.Trial_type_id
INNER JOIN
                  Varietal_trial ON Trial_series.Trial_series_id =
Varietal_trial.Trial_series_id INNER JOIN Var_trial_centre ON
Varietal_trial.Varietal_trial_id = Var_trial_centre.Varietal_trial_id INNER JOIN
[Zone] ON Varietal_trial.zone_id = [Zone].Zone_id ON State.State_id =
[Zone].State_id INNER JOIN Design ON Varietal_trial.Design_id =
Design.Design_id ON centre.Centre_id = Var_trial_centre.Centre_id
where  Trial_type.trial_type_name={?aa} and Trial_series.Trial_series_name={?bb}
and [Zone].Zone_name={?cc} and Varietal_trial.[Year]={?dd}
```

4.3 ISVE Database Relationships

The entity relationship (ER) diagram of ISVE is given in Figure 4.1.

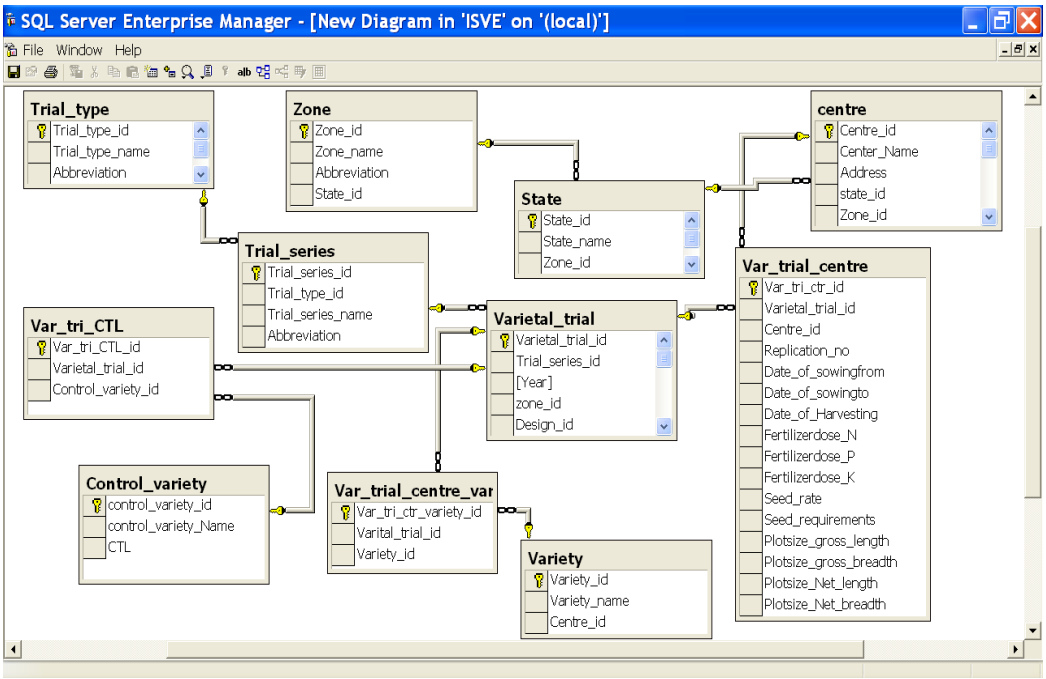

Fig. 4.1 ER Diagram of ISVE

CHAPTER V

Installation of ISVE

Before installing ISVE it is to be ensured that the computer has following facilities:

i) **Internet Information Services (IIS):** IIS version 5.0 or above is the web server required to run .NET web application to be deployed on any windows server. Copy of the Windows operating system CD is required for the installation and configuration.

ii) **A Modern Web Browser:** Any modern, standards-compliant browser can be used like **Internet Explorer** (5.5 and above).

iii) **The .NET Framework Redistributable:** The .NET Framework 2.0 or above is the basic requirement for this application to be installed. IIS 5.0 or above automatically supports ASP.NET (.aspx) pages to run on it.

iv) **The .NET Framework SDK:** The .NET Framework Software Development Kit (SDK) contains necessary Web application development tools, a debugger for error correcting, a development database engine in MSDE (Microsoft Data Engine), and a suite of samples and documentation.

v) **SQL Server** (Version 2000 and above)

vi) **Operating System Requirement:** A windows- based server Operating System such as Windows 2000 server, 2003 server or 2007 server is required for deployment of this application.

Other than that, enough disk space is required to install the Web server Internet Information Services (18 MB), the .NET Framework SDK (which includes ASP.NET; 108 MB), and a text editor like Notepad or Web Matrix or Visual Studio .NET.

5.1 Installing the Required Software

5.1.1 Installing Internet Information Services (IIS)

The database SQL Server 2000 or above is required as a backend. This can be installed on a server based operating system.

IIS comes with most versions of server-capable Windows operating systems, including Windows 2000 Professional, Server, and Advanced Server, Windows XP Professional, and Windows Server 2003, but it's not installed automatically in all versions. To see whether IIS is installed or not, it is required to navigate Administrative Tools menu and check if Internet Information Services is an option. Users of Windows 2000 Professional will find the Administrative Tools in their Control Panels, while XP and Server family users also have shortcuts in their start menus.

If the shortcut is not visible, then it is not installed. To install IIS, these steps are to be followed:

1. In the Control Panel, select Add or Remove Programs.
2. Choose Add/Remove Windows Components. The list of components will become visible within a few seconds.
3. In the list of components, check Internet Information Services (IIS).
4. Click Next. Windows prompts to insert the Windows CD and installs IIS.

Once IIS is installed, the Add or Remove Programs dialog is to be closed. One can check that IIS has installed correctly by seeing if it can be found within the Administrative Tools menu.

5.1.2 Installing Internet Explorer

For Windows user, Internet Explorer (IE) is installed by default, but recommendation is to use at least version 5.5. One can check the version by selecting About Internet Explorer from the Help menu.

If the version of Internet Explorer is earlier than 5.5, one can download the latest version free from the Internet Explorer Website (http://microsoft.com/windows/ie).

5.1.2 Installing the .NET Framework and SDK

The .NET Framework includes the necessary files to run and view ASP.NET (.aspx) pages, while the .NET Framework SDK includes samples, documentation, and a variety of free tools.

The .NET Framework is installed as part of the operating system of Windows Server 2003, in which case one can skip directly to installing the SDK. If not, it can be downloaded freely which includes the files necessary for running ASP.NET applications.

It is optional to install the software development kit, which includes necessary tools along with samples and documentation.

Warning

Installing the .NET Framework before installing IIS will prevent the applications from working correctly.

Downloading and Installing the Redistributable

The best method of acquiring the .NET Framework is to download and install it directly from the Web. To accomplish this, following steps are to be followed:

1. Go to the ASP.NET support site at http://www.asp.net/ and click the Download link.
2. Click the Download .NET Framework Redist Now link. It is to be remembered that the redistributable should be installed first, then the SDK. The link will advance to a download page.
3. Choose the language version of the install wanted, and click Download.
4. When prompted, save the file to a local directory by choosing Save.
5. After the download is complete, double-click the executable to begin the installation.
6. Follow the steps presented by the .NET Setup Wizard until installation completes.
7. .NET framework may also be downloaded from microsoft site or any other reliable website.

Downloading and Installing the SDK

After installing .NET framework, the software development kit (SDK) may also be installed. Following steps may be followed to install SDK:

1. Go to the ASP.NET support site at http://www.asp.net/ and click the Download link.
2. Click the Download .NET Framework SDK Now link. The link will advance to a download page.
3. Choose the language version of the install wanted to use and click Download.
4. When prompted to do so, save the file to a local directory by choosing Save.
5. After the download is complete, double-click the executable to begin the installation.
6. Follow the steps outlined by the .NET Setup Wizard until installation completes.

Once it's finished, one can check to see if it exists in programs menu by navigating to Start > Programs > Microsoft .NET Framework SDK.

5.1.3 Configuring IIS

Little configuration needs to be done before begining to work with IIS.

- Determining whether ASP.NET installed correctly
- Determining where files are located on the Web server
- Using localhost
- How to start and stop the Web server
- How to create a new virtual directory and modify its properties

5.1.4 Determining whether ASP.NET Installed Correctly

Once IIS is installed on the computer, one can open it by selecting Internet Information Services from the Administrative Tools menu. The first task is to make sure that ASP.NET was integrated into IIS when the .NET Framework is installed. Although, logically, ASP.NET should install automatically because it's a component of the .NET Framework, sometimes it doesn't. One can determine whether IIS was installed correctly by following these steps:

1. Open IIS, if it is not already done, and click on the + symbol next to the computer's name.
2. Right-click on Default Web Site and select Properties.

3. Navigate to the Documents tab. If default.aspx appears within the list, ASP.NET was installed correctly.

Another way to check whether ASP.NET installed correctly is by following these steps:

1. Navigate to the Application Mappings menu by right-clicking the root Website node (computer name) and choosing Properties.
2. Select the Home Directory tab, and choose Configuration.
3. The Application Mappings menu displays all of the extensions and their associated Internet Server Application Programming Interface (ISAPI) Extension Dynamically Linked Library(s) (DLLs), as in Figure 5.1.

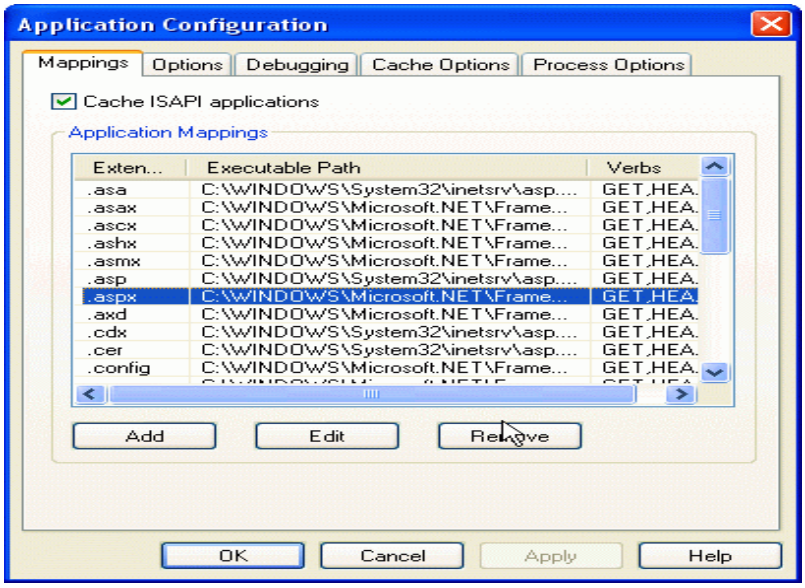

Figure 5.1. Application Mapping Menu

If the .aspx ISAPI Extension DLL appears within the Application Mappings menu, then ASP.NET was installed correctly.

If it is concluded that ASP.NET was not installed on the computer, it is to be installed manually from the command prompt:

1. Open the command prompt by selecting Start > Run, type CMD, and select OK.

2. Type the following command (all on one line) to install ASP.NET on Windows 2000 Professional, Server, or Advanced Server:
   C:\WINNT\Microsoft.NET\Framework\ver\aspnet_regiis.exe -i
   Or on Windows XP Professional:
   C:\WINDOWS\Microsoft.NET\Framework\ver\aspnet_regiis.exe -i
   In these commands, ver is the directory corresponding to the version of the .NET Framework installed.
3. Once ASP.NET is installed, close the command prompt and check again to confirm whether ASP.NET installed correctly.

If it still hasn't installed, it is advised to try visiting the Microsoft Knowledge Base (http://support.microsoft.com/) for help.

5.1.5 Location for putting files:

One can readily set IIS to look for Web applications within any folder of his/her choice, including the My Documents folder or even a network share. By default, IIS maps the wwwroot subfolder of C:\Inetpub on the server to Website's root directory, and it is generally considered a good repository for storing and managing Web applications.

If this wwwroot folder is opeded in Windows Explorer, and compareed with the folder tree that appears on the left of the IIS console, it is noticed that the folders in Explorer also appear under Default Web Site node. It is to be noted that while several of these folders have the regular Explorer folder icon in the IIS view, others have a special Web application icon, indicating that these folders contain the pages and other items for a particular Web application. These special folders are what IIS calls Virtual Directories, and, in fact, they do not have to share the name of the physical folder to which they map.

5.1.6 Using Localhost

By putting files within C:\Inetpub\wwwroot, Web server access is given to them. To open them in browser using the special Web address that indicates the current computer, http://localhost/. If it is tried, IIS will open up some HTML help documentation, because a default Website is yet to be set up. This localhost name is, in fact, equivalent to the so-called loopback IP address, 127.0.0.1, IP which one can

check out by entering http://127.0.0.1/ in the browser; one should see the same page that can be seen using localhost. If it is known, one can also use the name of the server or the real IP address of the machine to the same effect.

It is to be noted that if any of these equivalents is tried, a dialog will appear before the page is opened, asking for network credentials, because local authentication is no longer used implicit with localhost.

5.1.7 Stopping and Starting IIS

For the most part, it is always required to have IIS running, except when certain programs are used locally that open ports and allow intruders to compromise the security of the computer. Some programs, like Kazaa, automatically stop IIS upon launch, because of potential security vulnerabilities. To stop IIS when it's not being used, the steps outlined below are to be followed:

1. With IIS open, select Default Web Site. The Play, Stop, and Pause icons will become visible.
2. Select Stop, as shown in Figure 5.2.

Figure 5.2. Selection of the Stop icon to stop IIS.

3. To start IIS again, click on the Play icon.

5.1.8 Virtual Directories

A virtual directory is simply a name (or alias) that points to a local folder or network share on the server. This alias is then used to access the Web application held in that physical location. For instance, a company has a Web server that serves documents from C:\Inetpub\wwwroot\mySiteA. Users can access these documents through this URL:

http://www.mycompany.com/mySiteA/

One could also set up another physical location as a different virtual directory in IIS. If, for instance, somebody is developing another Web application, he/she could store the files for it in C:\dev\newSiteB. One could then create in IIS a

new virtual directory called, say, CoolPages, which maps to this location. This new site would then be accessible through this URL: http://www.mycompany.com/CoolPages/

As this application is in development, one would probably want to set IIS to hide this virtual directory from the public until the project is complete. Though existing Website would still be visible.

To create a virtual directory on the server now:

1. Right-click on Default Web Site and select Virtual Directory from the New submenu. The Virtual Directory Creation Wizard will appear. Click Next.
2. Type in an alias for the virtual directory, say, WebDocs. Click Next.
3. Browse for the directory in which the application is located. For this example, one can go to choose the My Pictures folder located within the My Documents directory. Click Next.
4. Set Access Permissions for the directory. Typically, one will want to check Read, Run scripts, and Browse. He/she will not need to select Write until getting into accessing the file system. Click Next.
5. Click Finish.

Once the new virtual directory has been created, it will appear within the Website list as shown in Figure 5.3.

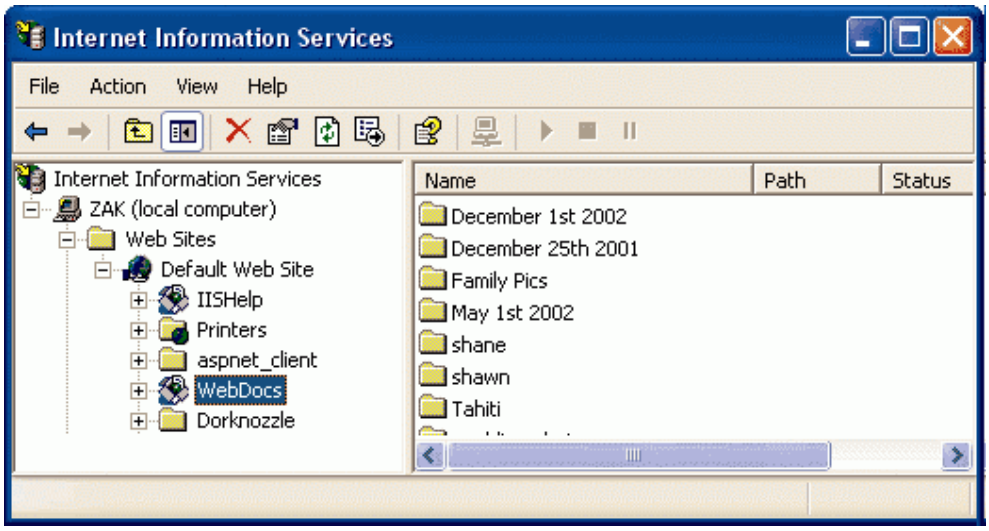

Figure 5.3. Virtual directory within the website.

Now, if it is typeed http://localhost/WebDocs/ in the browser, IIS will recognize that one is looking for a Website held in the My Pictures directory. By default, when one requests a virtual directory in this way, IIS looks for an index HTML page such as index.html or default.htm. If there is no index page—in this case there isn't—IIS assumes he/she wants to see the contents of the requested location. However, viewing the entire content of a location like this is not usually something wanted to do; all the files and directories that make up the Web page can be freely seen and accessed. Not only this is a little messy and unprofessional, but it also can provide information to hackers that could let them attack the site. So, by default, IIS won't allow this—a message "Directory Listing Denied" is received in the browser.

Bearing that in mind, there are, however, circumstances in which it is wanted to allow directory listings. To enable this in IIS:

(i) Right click the virtual directory in the IIS console, and choose Properties.

(ii) Select the Virtual Directory tab, and check the Directory browsing box. When it is clicked on OK and the same URL is opened (or refreshed) in the browser, a list of all the files within the My Pictures folder is seen.

The Properties dialog that is just used lets one configure various other useful properties, including:

Virtual Directory

Allows to configure directory-level properties including path information, virtual directory name, access permissions, etc. Everything that was set up through the wizard is modifiable through this tab.

Document

Allows to configure a default page that displays when the user types in a full URL. For instance, because default.aspx is listed as a default page, the user needs only to type in http://www.mysite.com/ into the browser's address bar, rather than http://www.mysite.com/default.aspx. One can easily change and remove these by selecting the appropriate button to the right of the menu.

Directory Security

Provides security configuration settings for the virtual directory.

HTTP Headers

Gives the ability to forcefully control page catching on the server, add custom HTTP Headers, Edit Ratings (helps identify the content the site provides to users), and create MIME types.

Custom Errors

Allows to define own custom error pages. Rather than the standard error messages that appear within Internet Explorer, one can customize error messages with company's logo and an error message of his/her choice.

One thing to note at this point is that one can set properties for the Default Web Site node, and choose to have them 'propagate' down to all the virtual directories created. So, it is required to enable directory browsing as the default for the Web applications. To enable directory browsing:

1. Right-click Default Web Site and select Properties. The Default Web Site Properties dialog will appear.
2. First, it is needed to remove the default setting which opens up the IIS help documentation for the root directory, so choose the Documents tab.
3. Select iisstart.asp, and click Remove.
4. Now choose the Home Directory tab.
5. Check the Directory Browsing check box and select OK.
6. When the Inheritance Overrides dialog appears, click Select All and then OK.

If it is typed http://localhost/ in the address bar of the browser, the directory listing will appear within the browser as shown in Figure 5.4.

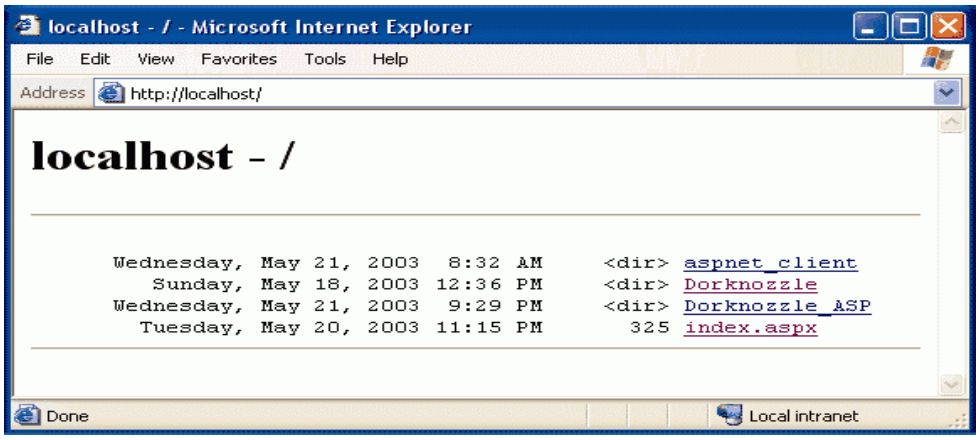

Figure 5.4. Enabling directory browsing for the Web server

Enabling directory browsing for the Web server provides with the ability to view directories in a way that's similar to the view within Windows Explorer.

5.2. Installing SQL Server

5.2.1 Software Requirements

SQL Server 2000 will run under Windows NT Server 4.0 with Service Pack 5 or later and Windows 2000 (all editions). SQL Server 2000 also requires Internet Explorer 5.0 or later, as it is required for Microsoft Management Console and HTML Help. Internet Explorer is not required to be the default browser and a minimal install is sufficient.

5.2.2 Installing SQL Server 2000

To Install SQL Server 2000 Basic Local Installation

1. Insert the Microsoft SQL Server 2000 compact disc in your CD-ROM drive (if the compact disc does not run automatically, double-click Autorun.exe in the root directory of the compact disc), select SQL Server 2000 Components, and then select Install Database Server. Setup prepares the SQL Server Installation Wizard. At the Welcome page, click Next.

2. In the Computer Name dialog box, Local Computer is the default option, and the local computer name appears in the text box. Click Next.

3. In the Installation Selection dialog box, click 'Create a new instance of SQL Server, or install Client Tools', and then click Next. Follow directions on the User

Information, Software License Agreement and related pages. In the Installation Definition dialog box, click Server and Client Tools, and then click Next.

4. In the Instance Name dialog box, if the Default check box is available, you can install either the default or a named instance. If the Default check box is not available, a default instance has already been installed, and you can install only a named instance.

a. To install the default instance, click to select the Default check box, and click Next.
b. To install a named instance, click to clear the Default check box, type a new named instance in the Instance Name box, and then click Next.

5. In the Setup Type dialog box, click Typical or Minimum, and then click Next.

6. In the Service Accounts dialog box, check the option ‘use the local system account’ in service settings part, and then click Next. In the Authentication Mode dialog box, accept the default setting, and then click Next. When you finish specifying options, click Next in the Start Copying Files dialog box.

7. In the Choose Licensing Mode dialog box, make selections according to your license agreement, and then click Continue to begin the installation. In the Setup Complete dialog box, click Yes; I want to restart my computer now, and then click Finish.

5.2.3 To Install Client Tools Only for SQL Server 2000

1. Insert the Microsoft SQL Server 2000 compact disc in your CD-ROM drive (if the compact disc does not run automatically, double-click Autorun.exe in the root directory of the compact disc), select SQL Server 2000 Components, select Install Database Server, and then click Next at the Welcome page of the SQL Server Installation Wizard.

2. In Computer Name dialog box, Local Computer is the default option, and the local computer name appears in the edit box. Click Next.

3. In the Installation Selection dialog box, click Create a new instance of SQL Server, or install Client Tools, and then click Next.

4. Follow the directions on the User Information, Software License Agreement, and related pages.

5. In the Installation Definition dialog box, click Client tools only, and then click Next.

6. In the Select Components dialog box, accept the defaults or select the components you want, and then click Next. You can select an item in the Components list, such as Management Tools, and then select items from the related Sub-Components list, such as Enterprise Manager. Click to select items that you want to install, and click to clear the check box for the items you do not want to install. For information about each component, select the item, and view the Description box.

7. In the Start Copying Files dialog box, click Next to complete the installation of the client tools.

5.2.4 To Install Connectivity Only for SQL Server 2000

1. Insert the Microsoft SQL Server 2000 compact disc into your CD-ROM drive (if the compact disc does not run automatically, double-click Autorun.exe in the root directory of the compact disc), and then select SQL Server 2000 Components.

2. Select Install Database Server. Setup prepares the SQL Server Installation Wizard. At the Welcome page, click Next.

3. In the Computer Name dialog box, Local Computer is the default option, and the local computer name appears in the text box. Click Next.

4. In the Installation Selection dialog box, click Create a new instance of SQL Server, or install Client Tools, and then click Next.

5. Follow the directions on the User Information, Software License Agreement and related pages

6. In the Installation Definition dialog box, click Connectivity Only, and then click Next.

7. In the Start Copying Files dialog box, click Next to complete the installation.

5.3 Installation and Execution of the Software

User has to start the computer and come to the default desktop of windows. He has to double click on My Computer icon. He has to Insert the disk of ISVE in computer and then double click on that disk drive icon. It will show the content of the disk

(All the files have been categorized into different folders under ISVE folder i.e. WebApplication folder and database folder). Then copy the WebApplication files in the C:\Inetpub\wwwroot directory. And restore the database in SQL Server after creating a new database. Then, the connection should be set according to the system in Web.config file in WebApplication, IIS etc.. After this, one has to start the server and then open the browser, enter following address in the address bar (http://localhost/ ISVE /Home.html). This will take user to the Home page of ISVE. Then navigate through the pages following the instructions given therein.

CHAPTER VI

Operations of ISVE

6.1 Introduction

Information System for varietal experiments is a web-based, user-friendly and integrated system. To start with it, open Internet Explorer and enter URL http://localhost/<project file name>/Home.aspx this will open the home page of the site with the IP address or <localhost> as shown in fig. 6.1.

Using home page users can view various topics of the system, such as, **"About ISVE**", **"Reports", "FAQ", "Contact Us", "Ask Us"** and **"Help".** The brief description of these can be seen by clicking on the link.

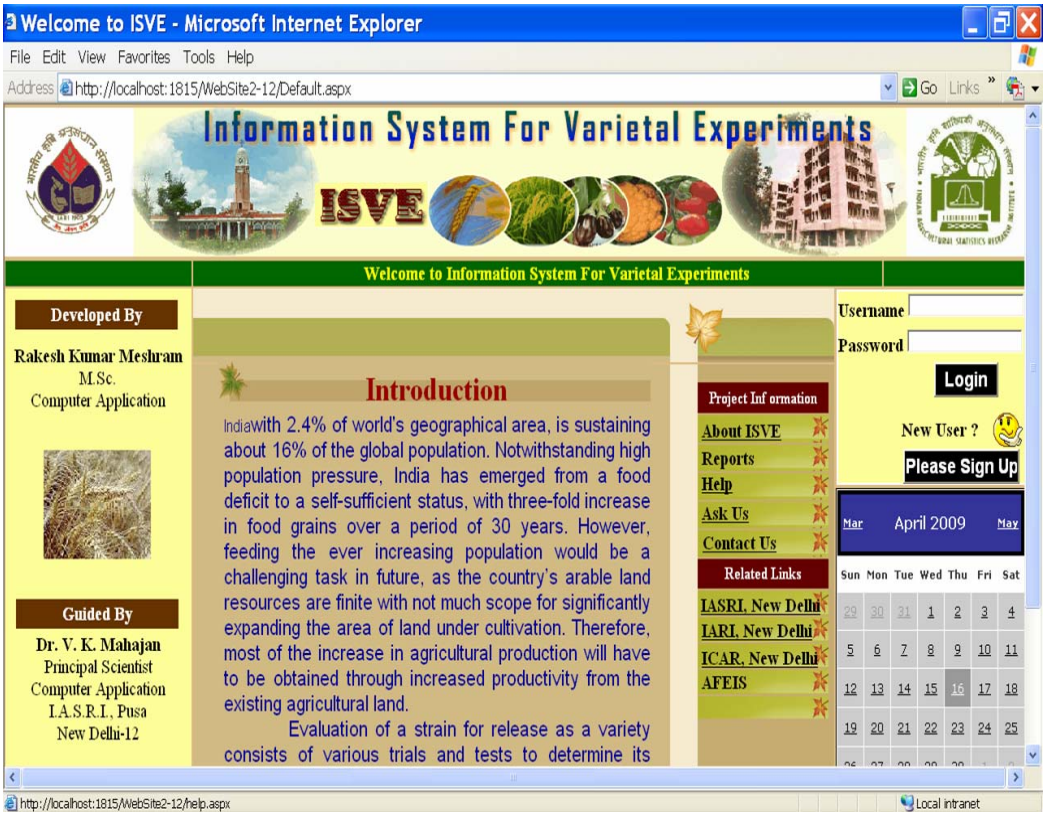

Fig.6.1 Home page for ISVE.

There are two types of users of the system – Administrator and other users. The system has been designed accordingly. The administrator requires "username" and "password" to enter the administrator page. The other users can browse the pages (except administrator pages) without username and password.

6.2 Administrator

Administrator is the person who has to manage the system, therefore, he has privilege to add, modify and delete information according to the need. An authorized login is necessary for modifying the database. The login screen for administrator is shown in fig. 6.1.

When the Administrator enters the correct username and password, he will be directed to the Data Management page for administrator. (Fig 6.2)

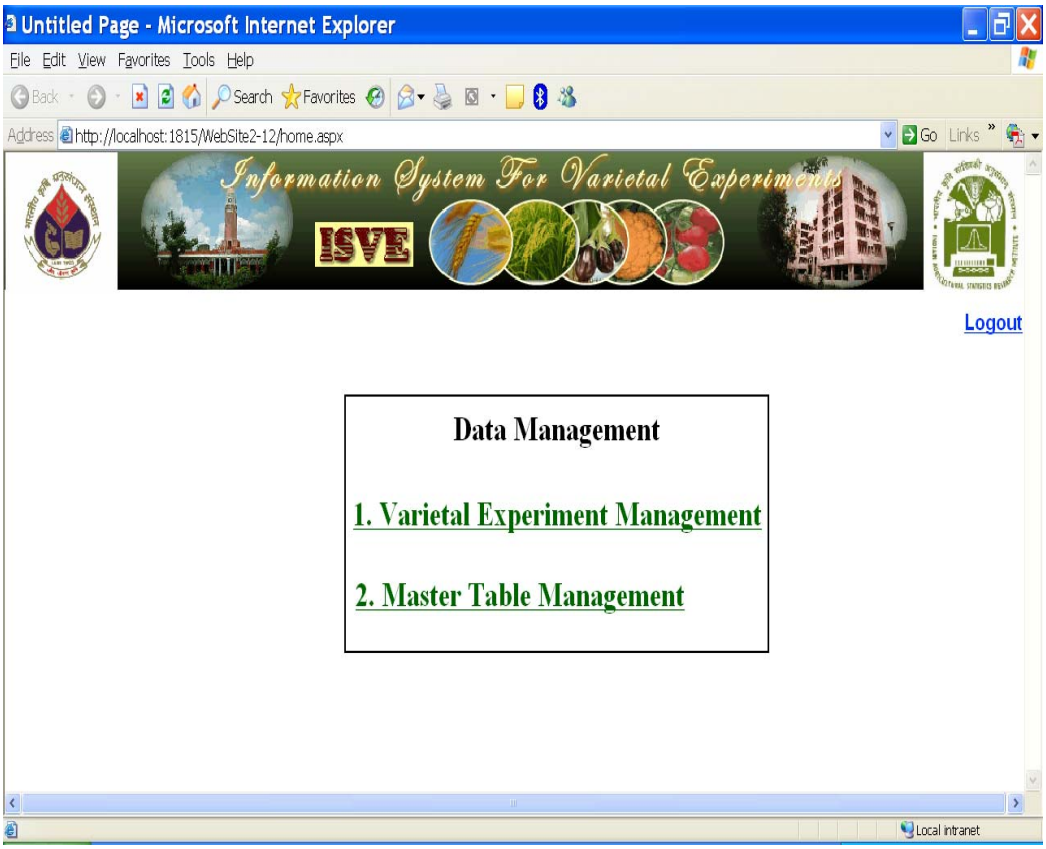

Fig. 6.2 Data Management

There are two options under Data Management which are Master Table Management and Varietal Experiment Management. Varietal Experiment Management is chosen only when Master Table Management is completed.

Master Table Management Page includes items such as Zone, State, Centre, Variety, Trial Series, Trial Type and Control Variety etc. Each option in master table management has facilities for addition of new records, modification of existing record and deletion of a record if not needed. After carrying out the desired work the system is to be logged out, for which a link “Logout” has been provided.

Fig. 6.3 Master table management.

6.3 Administrative Operations

Administrator can add, modify and delete any master table data. In master table data management page, links have been provided for each master table, from where administrator can choose any item for doing desired operation.

6.3.1 Addition of new items in master tables

In case of addition, administrator can add record for any of the master table mentioned. For example a screenshot of adding new state in state table is shown in fig. 6.4.

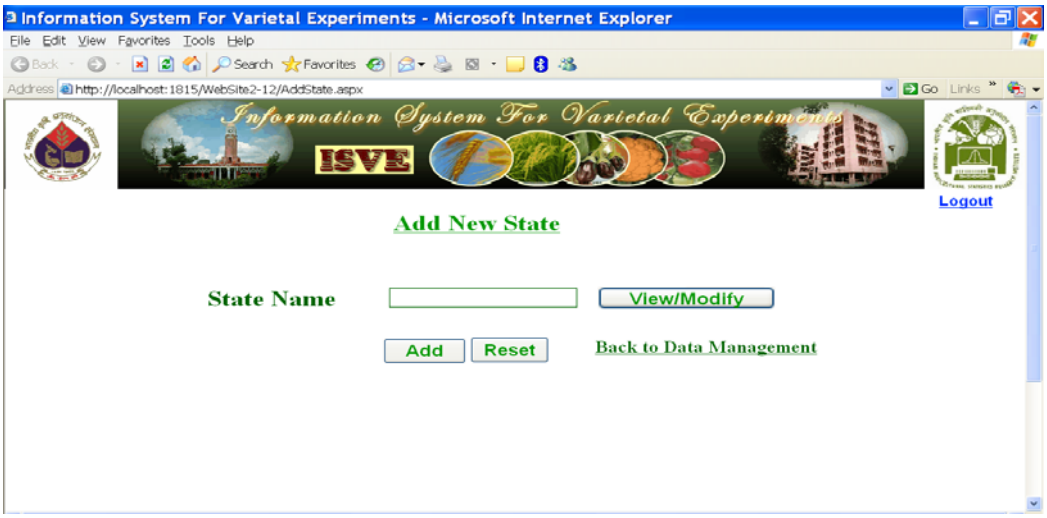

Fig: 6.4 Add new state.

If a state name already exists, an error message “State Already Exists” will be displayed on the screen as shown in fig. 6.5.

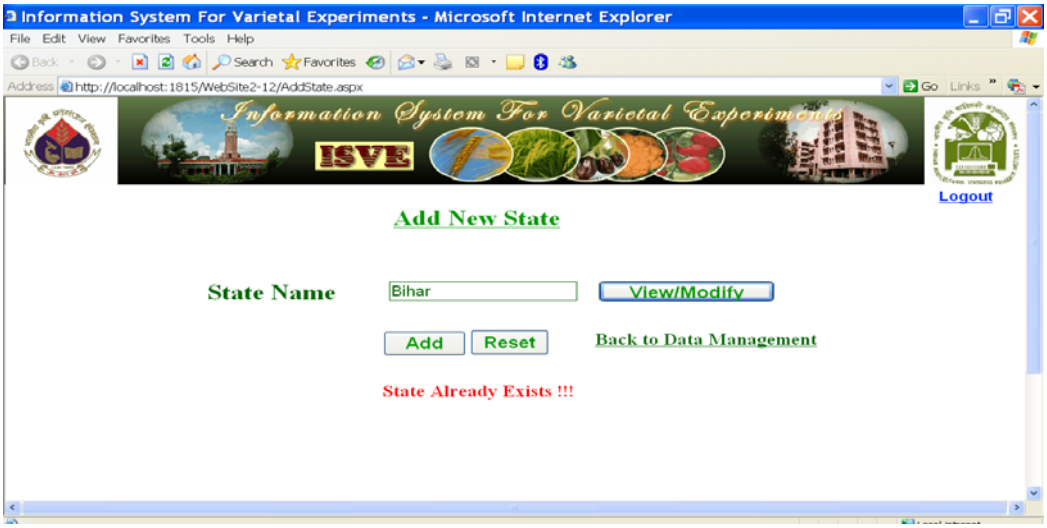

Fig: 6.5 State already exists.

After successful addition of a new state to the database, the system displays a message “State Successfully Added” as shown in fig. 6.6.

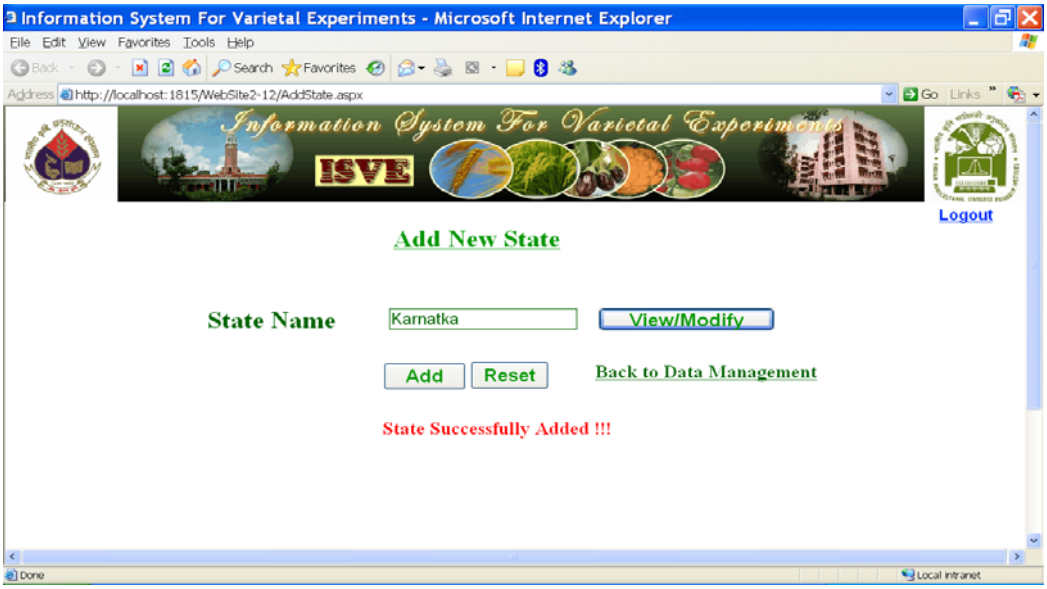

Fig: 6.6 Successful Addition of new state

6.3.2 Modification of existing records in master tables

Administrator can also modify a record in master table if there is any error in the existing record. For modifying a record, ‘View/Modify’ button has been provided, which is clicked after selecting that record from the record list. On clicking the button a page containing information about the selected record is displayed as shown

in fig. 6.7. After doing the desired modification 'update' button can be clicked for saving the modified information into the database.

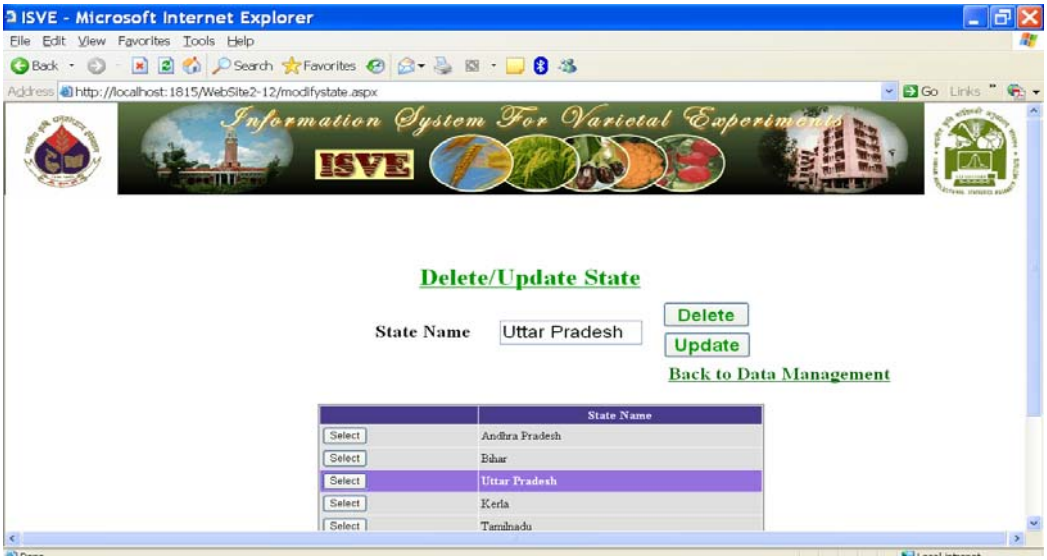

Fig: 6.7 Modification of existing state

6.3.3 Deletion

An unwanted or irrelevant record can also be deleted by the administrator from the database. He can click on the 'Delete' button for deletion of the selected record. On click of the "Delete" button, an alert message box appears with the message as asking "Are you sure, you want to delete the record". Administrator can click "Ok" button to confirm deletion or "Cancel" to cancel the deletion. (Fig. 6.9)

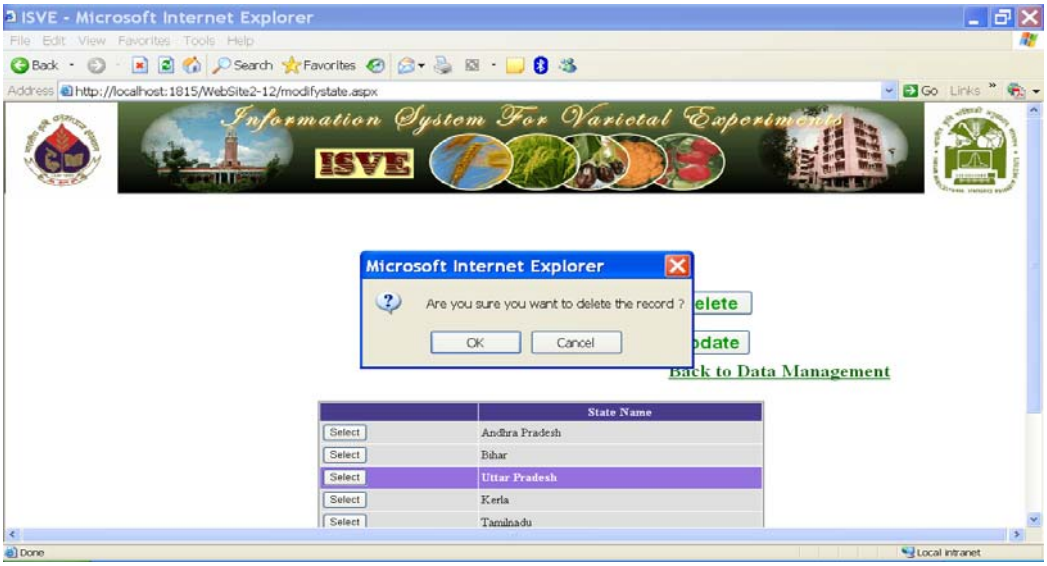

Fig: 6.8 Deletion of irrelevant record

After the confirmation, selected state will be deleted from the database and "State Successfully Deleted" message will be displayed. (Fig. 6.9)

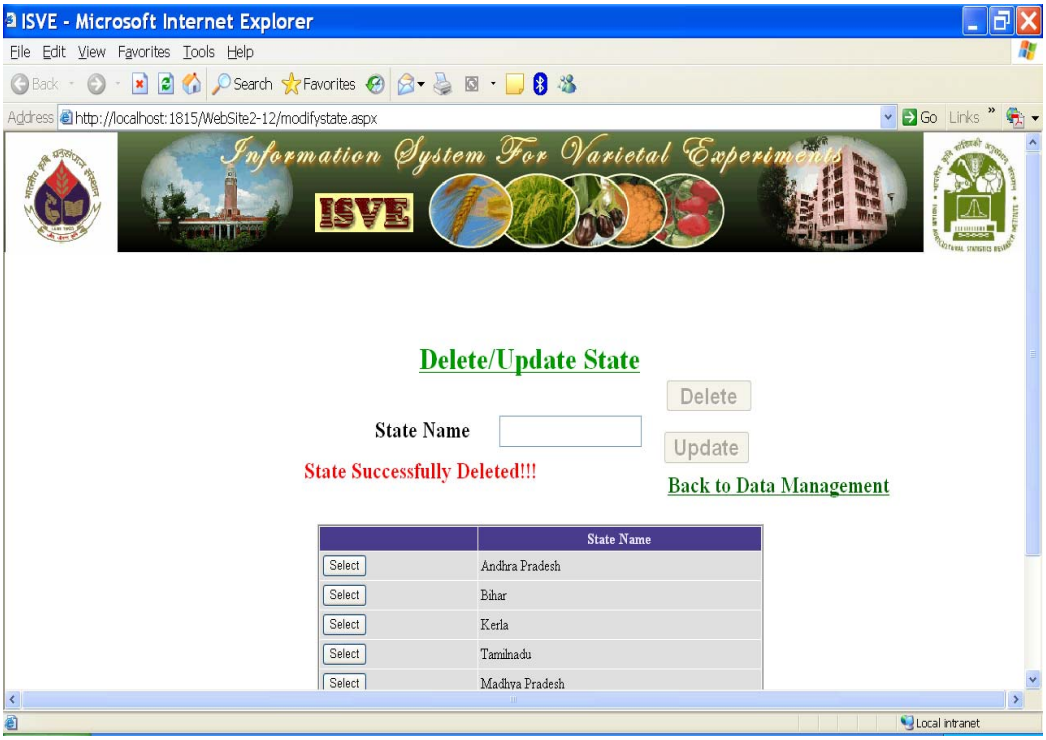

Fig: 6.9 Successful deletion.

Similarly, for other master tables' zone, centre, variety, trial series, trial type and control variety also, a similar facilities have been provided.

6.4 Varietal Experiment Management

After the completion of master tables management, administrator can click on the link "Varietal experiment management" on the Data Management page for entering details about varietal experiment.

Varietal experiment management is carried out in four steps as below:

1. Detail of varietal experiment (Add Centre).
2. Detail of varietal experiment (Add Varieties).
3. Detail of varietal experiment (Add Control varieties).
4. Varietal experiment data.

6.4.1 Detail of Varietal Experiment (Add Centre)

In this screen all centers conducting the trial is added along with other plot details. Once added it can be modified later. (Fig. 6.10)

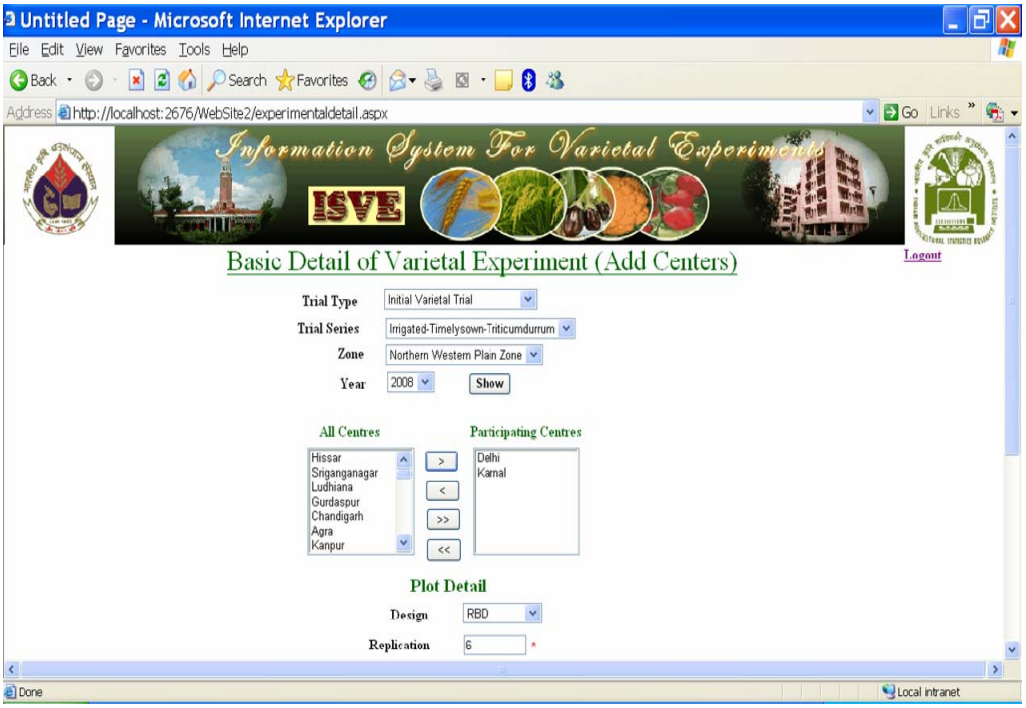

Fig: 6.10 Detail of varietal experiment (Add centre)

6.4.2 Detail of Varietal Experiment (Add Varieties)

Varieties which are to be tested can be added or deleted for the selected varietal experiment. (Fig. 6.11)

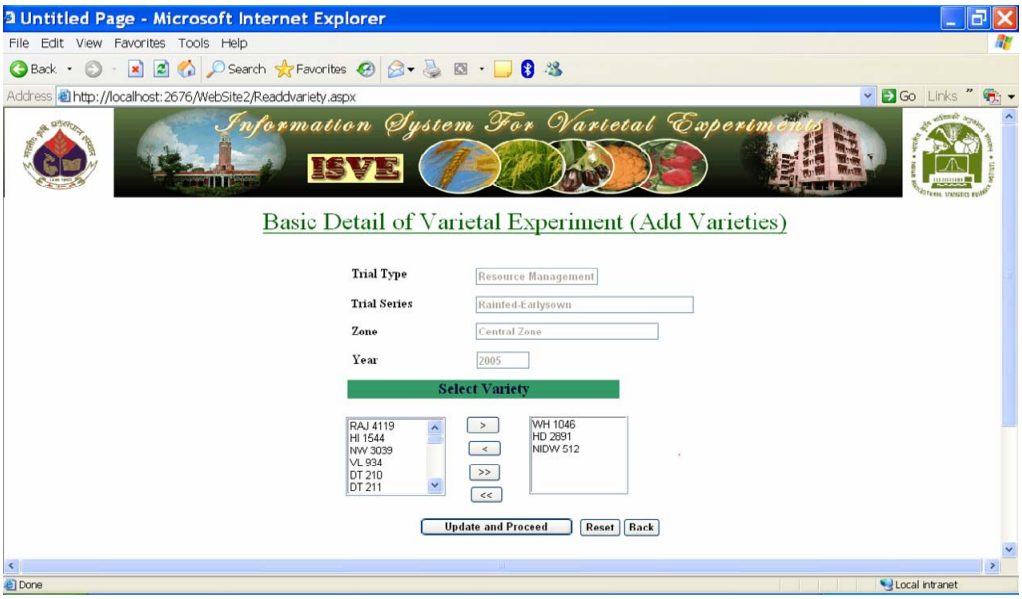

Fig: 6.11 Detail of varietal experiment (Add Varieties)

6.4.3 Detail of Varietal Experiment (Add Control Varieties)

Varieties for control against the new varieties can be added or deleted for the selected varietal experiment. (Fig. 6.12)

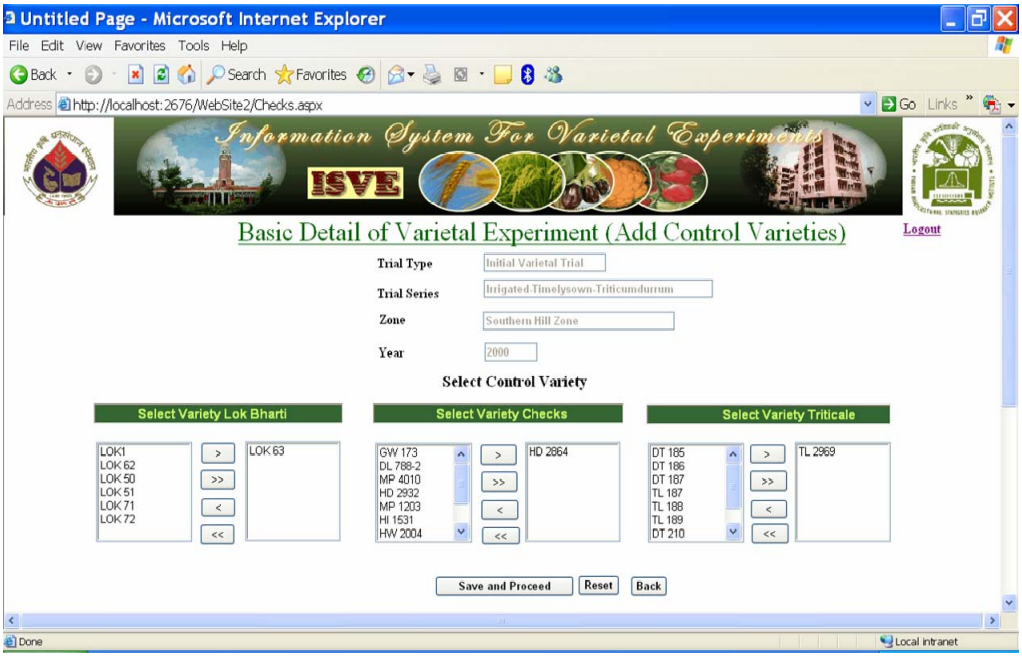

Fig: 6.12 Detail of varietal experiment (Add Control Variety)

6.4.4 Varietal experiment data

On the basis of above information one data sheet will be generated where the experimental data can be added or modified. (Fig. 6.13)

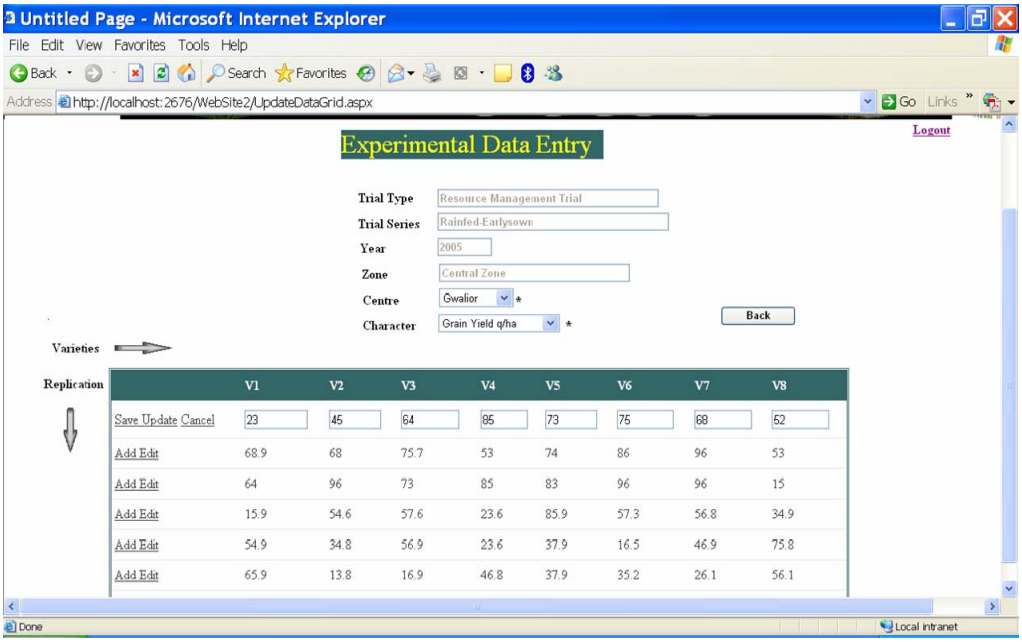

Fig: 6.13 Experimental data entry.

6.5 Report Generation

On Clicking ‘Reports’ link on the home page, user can view the report page. For report page no login is required. (Fig. 6.14)

Fig: 6.14 Reports.

6.5.1 Report on contributing centers for varieties

In report page, all the items for which reports can be generated are listed. User can select any item for generating report. After selecting any item from the list (For example, contributing centers for varieties), corresponding page will open where user can search centre name by selecting provided criteria. (Fig. 6.15)

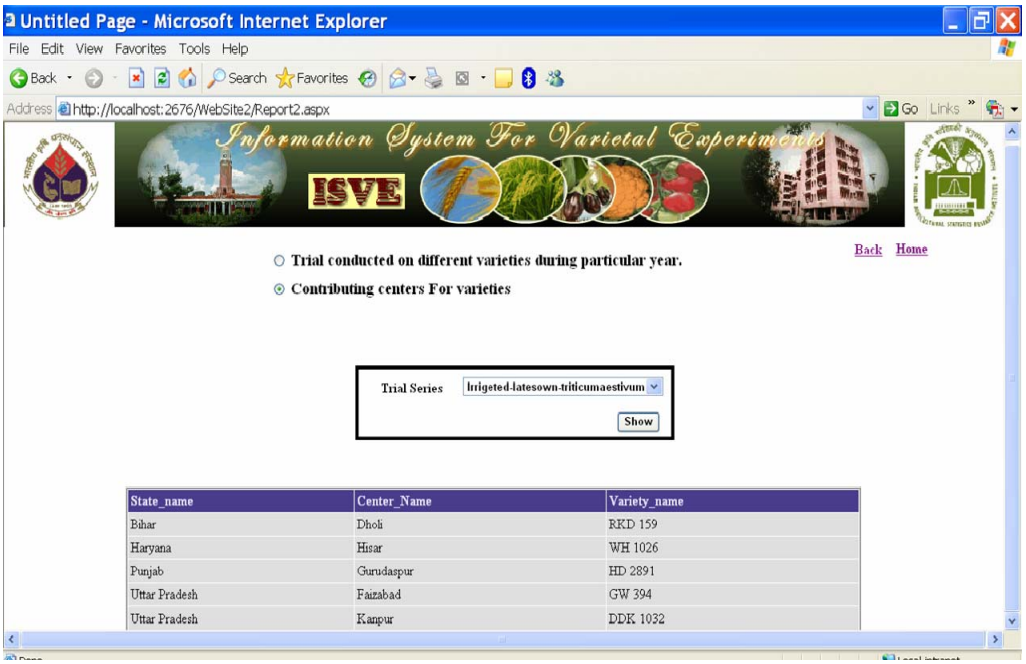

Fig: 6.15 Contributing centers for varieties

6.5.2 Report for experimental detail

Here, user can see the experimental detail such as trial type, trial series, zone, participating centers where experiment was conducted, contributing centers who contributed the variety and plot details etc. (Fig. 6.16)

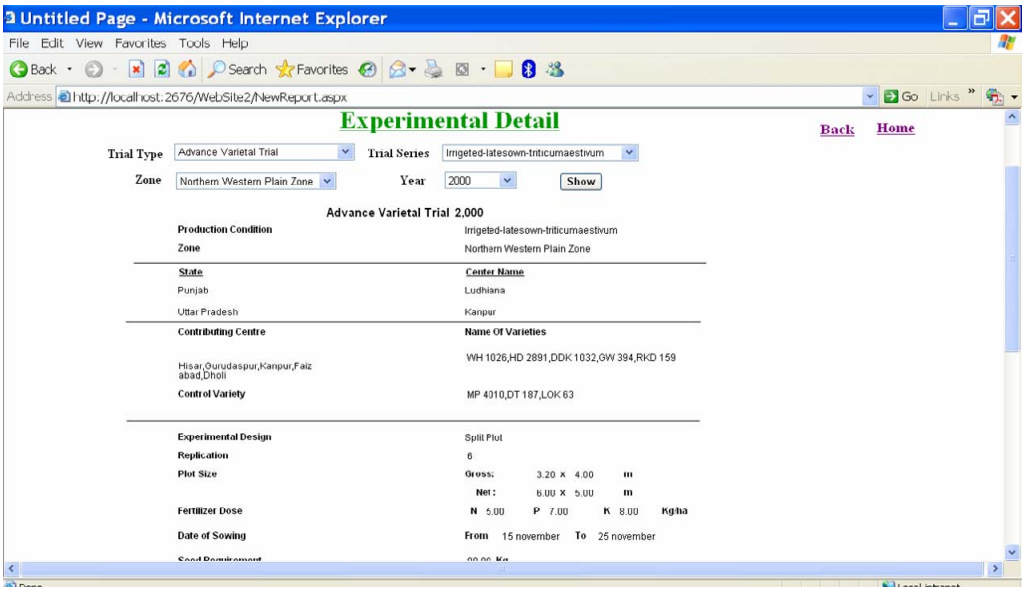

Fig: 6.16 Report for experimental detail

6.5.3 Report for data sheet

Here, user can see the data sheet for a particular experiment for different centers and characters. (Fig. 6.17)

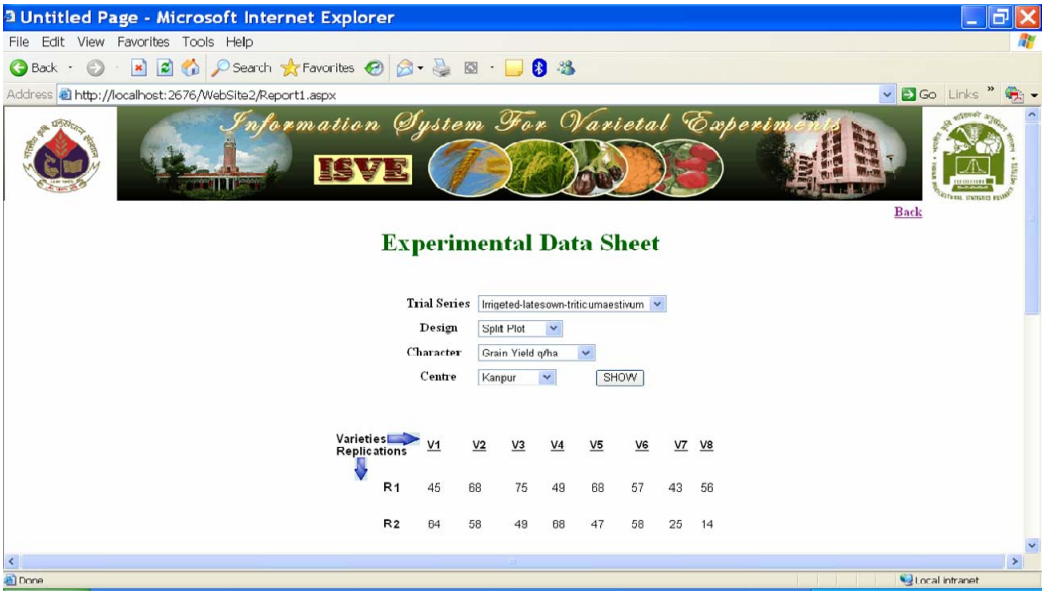

Fig: 6.17 Report for data sheet

6.6 Help

ISVE help page is shown below (Fig. 6.18).

Fig: 6.18 Help

6.7 Frequently Asked Questions

Users can view some of their queries by viewing the Frequently Asked Questions (FAQ) where frequently asked queries are displayed. (Fig.6.23).

Fig: 6.19 frequently asked questions

6.8 Contact Us

This section contains the name, address and e-mail of the various personalities related to this work. Users can click the desired e-mail id and directly send any query to that person with outlook express (Fig. 6.20).

Fig: 6.20 Contact Us

6.9 Ask Us

If user wants to send any questions /comments to Administrator, he can send it through “Ask Us” page. User needs to specify his e-mail address and question. (Fig. 6.21)

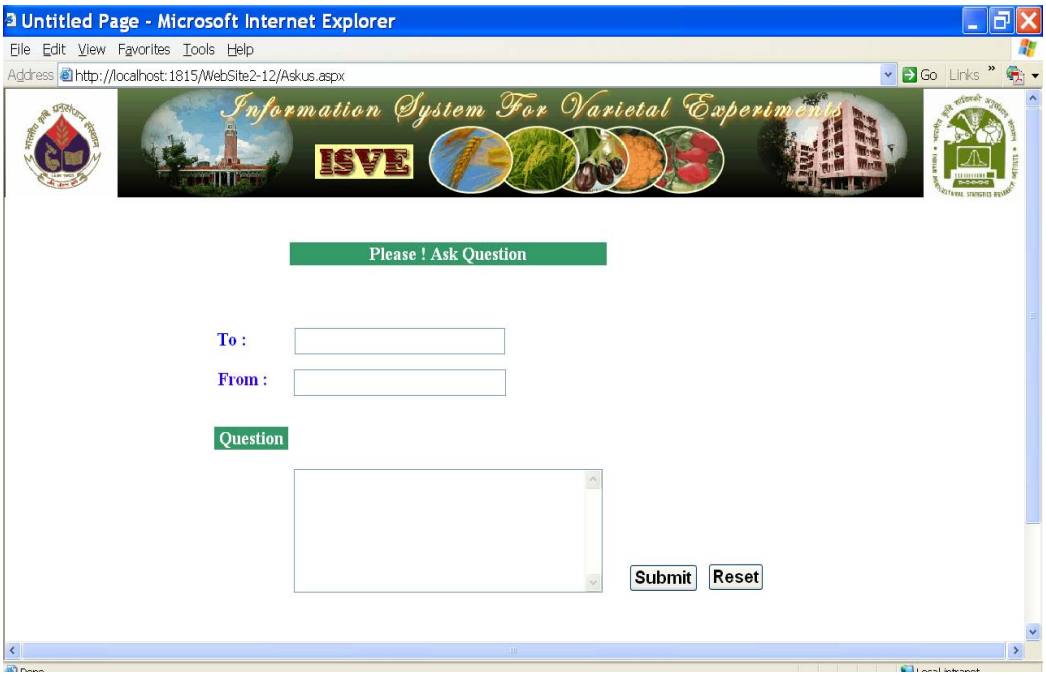

Fig: 6.21 Ask Us

6.10 About ISVE

In this page, some general information about the software is available. (Fig. 6.22)

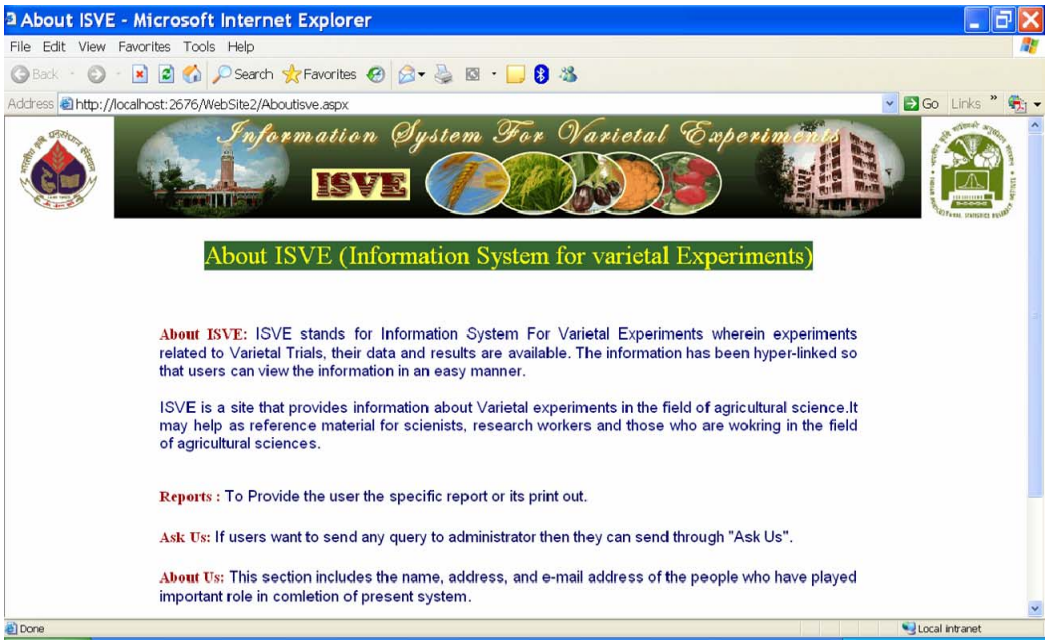

Fig: 6.22 About ISVE

CHAPTER VII

Future Scope of Study

No software can claim to provide complete solution for all problems of given type and for all time to come. If it would not be the case, every commercial software package would not have various successive improved versions after their first release. The present software is the first version and there is a scope for further enhancement. The following enhancements may be implemented in the ISVE in future.

- A data analysis module can be added.
- ISVE can be utilized for developing software relating to other crops.
- A module on '**On-line Chat**' can be built in so that end-users can interact with the each other and get required information.

Summary

The present era has seen an exponential growth and diversification in all forms of information management systems, which is sometimes called, as information explosion. It has become possible due to the impact of computer technology on the modern society. Over the last few decades, India has made impressive progress in the field of Agriculture. Agriculture still remains the most vital sector in the economy of our country.

India has made impressive progress in the field of agriculture. Even in the today's era of industrialization & communication, it still forms the backbone of Indian economy. Though agriculture is the vital sector of the economy of our country yet, it is one enterprise that faces uncertainty at every stage.

The India has a number of regional stations in diverse agro-ecological areas. The plant breeders' test/evaluate their varieties at these stations and generate voluminous information on the performance and stability of the varieties. Out of this experience the AICRP (All India Coordinated Crop Research Projects) was designed to share centers the elite breeding material, conduct identical yield trials to assess the best and stable genotype and their suitability for release as variety. On the strength of this working model in 1965, the All India Coordinated Wheat Improvement Project (AICWIP) was started to accelerate the process of varietal development. The project has active multidisciplinary research centers housed in the various SAUs and these centers get financial support to meet 75% of the annual expenditure from the federal government and the remaining from the state. This cost sharing brought about a cohesive, purpose and result oriented network to develop varieties and at annual meeting all the AICWIP researchers exchange the trial data, results etc.

The lack of information on the availability of best varieties is one of the main reasons for the limited use of information available at AICWIP. Sometimes, the farmers may not know the right variety, for particular time and specific locations. Since varieties are zone specific therefore, proper selection of the variety for specific location is very important for better yield.

Computerized information systems have influenced nearly all types of organizations, whether small or large, public or private, national or multinational. Information systems exist for almost all the fields may be Library management, Farm Management, Industry Management, Satellite Management, Decision Support System, Expert System etc. So, the basic purpose of this information system is to provide the relevant information to extension personnel, students and researchers which help them in decision making related to selecting right varieties, their locations and their production conditions for better yield.

The proposed software package is being developed to fulfill the above objectives and provide an easy access to the complete relevant information related to various institutes and projects under different divisions of ICAR, which are relevant from the management point of view. A database can be created, by using data entry option of the software package for varietal experiments. Finally, the data can be merged to provide a comprehensive management information system.

It has a three-layered architecture. Client Side Interface Layer is implemented using HTML and JavaScript, Server Side Application Layer using ASP.NET and C# for Database Connectivity. Database Layer is implemented using SQL server. **ISVE** can be implemented as a network-based system with a server at a central location (IASRI) so that information is available on-line. **ISVE** runs at any node of the Internet through a browser. Security features are provided in such a way that only concerned person can access the database. There is provision to insert, update and delete the information. **ISVE** provides query & report generation facility also. User can interact with subject specialists through email. On-line help scheme is provided to help administrators and users both. The feature of providing information to users through frequently asked questions has also been included in this software.

Information System for Varietal Experiments

Abstract

The present era has seen an exponential growth and diversification in all forms of information, sometimes called, an information explosion. It has been made possible due to the impact of computer technology on the modern society. Computerized information systems have influenced nearly all domains. The domain of Agriculture is no exception.

India has made impressive progress in the field of agriculture but, it is one enterprise that faces uncertainty at every stage. India has a number of regional stations in diverse agro-ecological areas. The plant breeders' test/evaluate their varieties at these stations and generate voluminous information on the performance and stability of the varieties. Out of this experience the AICRP (All India Coordinated Crop Research Projects) was designed to share centers the elite breeding material, conduct identical yield trials to assess the best and stable genotype and their suitability for release as variety. On the strength of this working model in 1965, the All India Coordinated Wheat Improvement Project (AICWIP) was started to accelerate the process of varietal development. But the lack of information on the availability of best varieties is one of the main reasons for its limited use. Sometimes, the farmers may not know the right variety, right time and right locations etc. Because of this, farmers don't get expected results after using it. Varieties are zone specific and therefore, proper selection of the varieties and right locations are very important for better yield.

Information System for Varietal Experiments (ISVE) is a Web-based Information System to provide information to extension personnel, students, researchers etc. ISVE has a simple query and report generation module to provide the information about zone, centers, variety, trial type, trial series etc. even in the printable formats.

The software has one level of authentication i.e. administrator. Administrator has the privilege to add, modify or delete information from the database. Users are free to get information using this software. Users can ask questions regarding any information or about the software to the concerned experts by sending an e-mail; this

facility is included in the software itself. Users can also view some frequently asked questions (FAQs).

ISVE is developed using ASP.NET. It is a new web-based technology in the scenario. It is an easy and effective tool to develop web based applications. Database part is developed using SQL Server 2000. It is the database widely used for its simplicity and ease in operation.

उपजातीय परीक्षणों के लिए सूचना तंत्र

सार

वर्तमान युग में सूचना के सभी रूपों में निरंतर विकास एवं विविधतता देखी गयी है जिसे कई बार सूचना विस्फोट भी कहा जाता है । ऐसा आधुनिक समाज में संगणक प्रौद्योगिकी के प्रभाव से सम्भव हो सका है। कम्प्यूटरीकृत सूचना तंत्र ने लगभग सभी क्षेत्रों को प्रभावित किया है । कृषि का क्षेत्र कोई अपवाद नहीं है ।

भारत में कृषि के क्षेत्र में प्रभावशाली प्रगति हुई है परन्तु यह ऐसा उद्यम है जिसमें प्रत्येक चरण में अनिश्चितता पायी जाती है । भारत में विविध कृषि–पारिस्थितिकीय क्षेत्रों में अनेक क्षेत्रीय केन्द्र हैं। पादप प्रजनक इन केन्द्रों में अपनी किस्मों का परीक्षण/मूल्यांकन करते हैं और इन किस्मों के कार्य निष्पादन व स्थायित्वता पर वृहत् सूचना तैयार करते हैं । इस अनुभव से केन्द्र की विशिष्ट प्रजनन सामग्री को शेयर करने, श्रेष्ठ एवं स्थायी जीनोटाइप का मूल्यांकन करने हेतु समरूपी उपज परीक्षणों का संचालन करने तथा उन्हें किस्म के रूप में जारी करने हेतु उपयुक्तता का पता लगाने के लिए ए.आई.सी.आर.पी. (अखिल भारतीय समन्वित फ़सल अनुसंधान परियोजनाएँ) तैयार की गयी। सन 1965 में इस वर्किंग मॉडल की स्ट्रेन्थ पर, किस्मीय विकास की प्रक्रिया में तेजी लाने के लिए अखिल भारतीय गेहूँ संवर्धन समन्वित परियोजना (ए.आई.सी.डब्ल्यू.आई.पी.) आरम्भ की गयी। परन्तु इसके सीमित उपयोग का मुख्य कारण श्रेष्ठ किस्मों पर सूचना के उपलब्धता में कमी है । कई बार कृषकों को सही किस्म, सही समय तथा सही स्थान इत्यादि का ज्ञान नहीं होता । इस कारण इसके उपयोग के बाद भी किसानों को प्रत्याशित परिणाम प्राप्त नहीं होते । किस्में क्षेत्र आधारित होती हैं अतः किस्मों का सही चयन तथा सही स्थान उच्च उपज के लिए बहुत महत्वपूर्ण हैं ।

उपजातीय परीक्षणों के लिए सूचना तंत्र (आई.एस.वी.ई.) एक वेब–आधारित सूचना तंत्र है जो विस्तार कार्मिकों, छात्रों, शोधकर्ताओं इत्यादि को सूचना उपलब्ध कराता है । क्षेत्र, केन्द्र, किस्म, परीक्षण प्रकार, परीक्षण सिरीज इत्यादि के सम्बन्ध में सूचना उपलब्ध कराने के लिए आई.एस.वी.ई. का एक साधारण प्रश्न और रिपोर्ट जनरेशन मॉड्यूल है तथा इससे सूचना को प्रिण्ट भी किया जा सकता है ।

इस सॉफ़्टवेयर में प्राधिकरण का एक स्तर है जो ऐडमिनिस्ट्रेटर होता है । ऐडमिनिस्ट्रेटर को डाटाबेस में सूचना जोड़ने, संशोधित करने अथवा मिटाने की सुविधा उपलब्ध है । प्रयोक्ता को सॉफ़्टवेयर से सूचना प्राप्त करने की छूट होती है । वे सम्बन्धित विशेषज्ञों से ई–मेल के माध्यम से कोई भी सूचना अथवा सॉफ़्टवेयर के सम्बन्ध में कोई भी जानकारी प्राप्त कर सकते हैं । यह सूचना सॉफ़्टवेयर में ही शामिल है । प्रयोक्ता आमतौर पर पूछे जाने वाले प्रश्नों (एफ.ए.क्यू) को भी देख सकते हैं ।

ASP.NET के उपयोग से आई.एस.वी.ई.. विकसित किया गया है । यह इस परिवेश में एक नई वेब–आधारित प्रौद्योगिकी है । वेब–आधारित एप्लीकेशन्स विकसित करने के लिए यह एक आसान एवं प्रभावशाली माध्यम है । डाटाबेस SQL Server 2000 के उपयोग द्वारा विकसित किया गया है। इस डाटाबेस के साधारण एवं संचालन में आसान होने के कारण इसका व्यापक उपयोग है ।

References

Ahmad, Md. Z. (2000). *Information system on apple crop.* Unpublished M.Sc. Thesis, I.A.R.I., New Delhi.

Chadha, K. L. (2001). *Handbook of Agriculture*. ICAR Publications, New Delhi. *pp*: 1-21.

Das, S. (2000). *Web based Information System on Integrated Pest Management of Cucurbits.* Unpublished M.Sc. Thesis, I.A.R.I., New Delhi

Date, C. J. (2003) *Introduction to Database Systems* (7th Edition), Addition-Wesley/ Narosa, New Delhi.

Esposito, D. (2005). *Programming Microsoft ASP.NET 2.0,* WP Publishers & Distributers (P) Limited.

Evjen, B., Hanselman, S., Rader, D., Muhammad, F. and Shivakumar, S. (2006). *Professional ASP.NET 2.0*, Special Edition, Wiley India (P) Ltd., New Delhi.

Gogoi, A. K., Yaduraju, N.T., Dubey, R. P., Mishra, J. S. (2003). *Local names of Weeds.* National Research Centre for Weed Science, Maharajpur, Jabalpur 482 004 (MP).

Gruber, M. (2005). *SQL Instant Referenc* .Second Edition, BPB Publication,New Delhi.

Jana, C. (2008). *Information System on Herbicide use in Field Crops.* Unpublished M.Sc. Thesis, I.A.R.I., New Delhi

Kumar, R. J.,Jagannathan, R. (2003). *Weed Science Principles*. Kalyani Publishers, New Delhi-110002.

Kumari, A. (2005). *Information System for Major Fruit crops (Apple, Banana & Mango) of India.* Unpublished M.Sc. Thesis, I.A.R.I., New Delhi.

MacDonald, M. (2006). *The Complete Reference ASP.NET*. Tata McGraw-Hill Publishing Company Limited, New Delhi.

Prashant, P. (2000). *Pest Management Information System on Tomato.* Unpublished M.Sc. Thesis, I.A.R.I., New Delhi.

Pal, S. (2005). *Decision Support System for Nutrient Management in Crops.* Unpublished M.Sc. Thesis, I.A.R.I., New Delhi.

Pelland, P.(2006). *Build a Program Now! Microsoft Visual Basic 2005 Express Edition*. Prentice, Hall of India Private Ltd. New Delhi-110001.

Progress Report (2006-07). *Crop Improvement*, Directorate of Wheat Research Karnal.

Progress Report (2006-07). *Resource Management*, Directorate of Wheat Research Karnal.

Saurabh, C.(2005) *Intercropping experiments information system*. Unpublished M.Sc. Thesis, I.A.R.I., New Delhi.

Singh, B.D., *Plant Breeding 2000*, Sixth Edition, Kalyani Publishers, New Delhi.

Ullman, C., Kauffman, J., Hart, C., Sussman, D. and Maharry, D. (2005). *Beginning ASP.NET 1.1 with Visual C#.NET 2003*, First Edition, Wiley Dreamtech India (P) Ltd., New Delhi.

Walther, S. (2006). *ASP.NET 2.0 Unleashed*. Dorling Kindersley (India Pvt. Ltd.).

http://support.realsoftware.com/listarchives/gettingstarted/2004-05/msg00194.html

http://www.codeproject.com

http://www.asp101.com

http://www.c-sharpcorner.com

http://www.startC#dotnet.com/aspsite/controls/listbox.aspx

Appendix

Sample Code

Code for Home Page

```
<**********************************************************************>

<%@ Page Language="C#" AutoEventWireup="true"
CodeFile="Default.aspx.cs" Inherits="_Default" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" >
<head runat="server">
    <title>Home Page</title>
<script language="javascript" type="text/javascript">
// <!CDATA[

function IMG1_onclick() {

}

// ]]>
</script>
</head>
<body>
    <form id="form1" runat="server">
    <div>

        <asp:HiddenField ID="HiddenField1" runat="server" />
        <asp:HiddenField ID="HiddenField2" runat="server" />
        <asp:HiddenField ID="HiddenField3" runat="server" />

    </div>
        <table style="z-index: 101; left: -1px; width: 1258px;
position: absolute; top: 0px; background-color: #debb84;">
            <tr>
                <td colspan="1" style="width: 227px; height: 142px">
                </td>
                <td colspan="5" style="height: 142px">
                    <img height="141"
src="NewFolder2/finallast11.gif" style="z-index: 100; left: 1px;
                        position: absolute; top: 3px"
width="1257"alt="text" />
                </td>
            </tr>
            <tr>
                <td colspan="1" style="width: 227px; height: 21px;
background-color: #006600; text-align: center">
                </td>
                <td colspan="4" style="height: 21px; background-
color: #006600; text-align: center">
                    <strong><span style="font-size: 14pt; color:
#ffff00"><marquee style="text-align: center">Welcome to Information
System For Varietal Experiments</marquee></span></strong></td>
                <td style="width: 173px; background-color: #006600;
font-size: 12pt;">
                </td>
```

```
            </tr>
            <tr style="font-size: 12pt">
                <td colspan="1" rowspan="5" style="width: 227px;
background-color: #ffff99; text-align: center;">
                    <strong>Rakesh Kumar Meshram<br />
                    </strong>M.Sc.<br />
                    Computer Application<br />
                    <br />
                    <br />
                    <br />
                    <br />
                    <br />
                    <br />
                    <br />
                    <br />
                    <br />
                    <br />
                    <br />
                    <strong>Dr. V. K. Mahajan<br />
                    </strong>Principal Scientist<br />
                    Computer Application<br />
                    I.A.S.R.I., Pusa<br />
                    New Delhi-12</td>
                <td colspan="3" style="height: 68px; background-
image: url(New Folder3/sub_header.jpg);">
                     
                    <table style="z-index: 103; left: 14px; width:
200px; position: absolute; top: 201px;
                        height: 24px; background-color: #663300">
                        <tr>
                            <td style="text-align: center">
                                <strong><span style="color:
#ffffff">Developed By</span></strong></td>
                        </tr>
                    </table>
                    <table style="z-index: 104; left: 15px; width:
200px; position: absolute; top: 468px;
                        height: 24px; background-color: #663300">
                        <tr>
                            <td style="text-align: center">
                                <span style="color:
#ffffff"><strong>Guided By</strong></span></td>
                        </tr>
                    </table>
                    <table style="z-index: 108; left: 852px; width:
165px; position: absolute; top: 261px;
                        height: 136px">
                        <tr>
                            <td style="height: 27px; background-
color: #660000; text-align: center">
                                <strong><span style="color:
#ffffff">Project Inf
                                    <asp:HyperLink ID="HyperLink7"
runat="server" Font-Bold="True" Font-Size="14pt"
                                    ForeColor="Black"
NavigateUrl="~/Aboutisve.aspx" Style="z-index: 100; left: 7px;
                                    position: absolute; top:
37px">About ISVE</asp:HyperLink>
                                    ormation</span></strong></td>
                        </tr>
                        <tr>
```

```
                            <td style="background-image: url(New
Folder3/linkpanel_hover.jpg); height: 24px;
                                text-align: center">
                                <strong style="behavior:
url(Admin.aspx); cursor: hand"> </strong></td>
                        </tr>
                        <tr>
                            <td style="background-image: url(New
Folder3/linkpanel_hover.jpg); height: 23px">
                                <asp:HyperLink ID="HyperLink8"
runat="server" Font-Bold="True" Font-Names="Times New Roman"
                                    Font-Size="14pt" Style="z-index:
100; left: 7px; position: absolute; top:
63px">Reports</asp:HyperLink>
                            </td>
                        </tr>
                        <tr>
                            <td style="background-image: url(New
Folder3/linkpanel_hover.jpg); height: 23px">
                                <asp:HyperLink ID="HyperLink3"
runat="server" Font-Bold="True" Font-Size="14pt" ForeColor="#000000"
                                    NavigateUrl="~/Askus.aspx"
Style="z-index: 100; left: 7px; position: absolute;
                                    top: 115px">Ask
Us</asp:HyperLink>
                            </td>
                        </tr>
                        <tr>
                            <td style="background-image: url(New
Folder3/linkpanel_hover.jpg); border-top-style: none;
                                border-right-style: none; border-
left-style: none; height: 23px; border-bottom-style: none; behavior:
url(help.aspx); cursor: hand; text-align: center;">
                                <strong><span style="font-size:
14pt; color: #006666">
                                    <asp:HyperLink ID="HyperLink1"
runat="server" Font-Names="Times New Roman" Font-Size="14pt"
                                        ForeColor="Black"
NavigateUrl="~/help.aspx" Style="z-index: 100; left: 7px;
                                        position: absolute; top:
87px">Help</asp:HyperLink>
                                </span></strong>
                            </td>
                        </tr>
                        <tr>
                            <td style="background-image: url(New
Folder3/linkpanel_hover.jpg); border-top-style: none;
                                border-right-style: none; border-
left-style: none; height: 23px; border-bottom-style: none">
                                <asp:HyperLink ID="HyperLink2"
runat="server" Font-Bold="True" Font-Names="Times New Roman"
                                    Font-Size="14pt"
ForeColor="Black" NavigateUrl="~/Contactus.aspx" Style="z-index:
100;
                                    left: 8px; position: absolute;
top: 143px">Contact Us</asp:HyperLink>
                            </td>
                        </tr>
                        <tr>
                            <td style="border-top-style: none;
border-right-style: none; border-left-style: none;
```

```
                                height: 26px; background-color:
#660000; text-align: center; border-bottom-style: none">
                                <strong><span style="color:
#ffffff">Related Links</span></strong></td>
                        </tr>
                        <tr>
                            <td style="background-image: url(New
Folder3/linkpanel_hover.jpg); border-top-style: none;
                                border-right-style: none; border-
left-style: none; height: 21px; text-align: center;
                                border-bottom-style: none">
                                <asp:HyperLink ID="HyperLink4"
runat="server" Font-Bold="True" Font-Names="Times New Roman"
                                    Font-Size="12pt"
ForeColor="Black" NavigateUrl="http://www.iasri.res.in/" Style="z-
index: 100;
                                    left: 8px; position: absolute;
top: 202px" Width="125px">IASRI, New Delhi</asp:HyperLink>
                            </td>
                        </tr>
                        <tr>
                            <td style="background-image: url(New
Folder3/linkpanel_hover.jpg); border-top-style: none;
                                border-right-style: none; border-
left-style: none; height: 21px; text-align: center;
                                border-bottom-style: none">
                                <asp:HyperLink ID="HyperLink5"
runat="server" Font-Bold="True" Font-Size="12pt" ForeColor="Black"

NavigateUrl="http://www.iari.res.in/" Style="z-index: 100; left:
7px; position: absolute;
                                    top: 226px">IARI, New
Delhi</asp:HyperLink>
                            </td>
                        </tr>
                        <tr>
                            <td style="background-image: url(New
Folder3/linkpanel_hover.jpg); border-top-style: none;
                                border-right-style: none; border-
left-style: none; height: 21px; text-align: center;
                                border-bottom-style: none">
                                <asp:HyperLink ID="HyperLink6"
runat="server" Font-Bold="True" Font-Size="12pt" ForeColor="Black"

NavigateUrl="http://www.icar.org.in/" Style="z-index: 100; left:
7px; position: absolute;
                                    top: 252px">ICAR, New
Delhi</asp:HyperLink>
                            </td>
                        </tr>
                        <tr>
                            <td style="background-image: url(New
Folder3/linkpanel_hover.jpg); border-top-style: none;
                                border-right-style: none; border-
left-style: none; height: 21px; text-align: center;
                                border-bottom-style: none">
                            </td>
                        </tr>
                        <tr>
                            <td style="background-image: url(New
Folder3/linkpanel_hover.jpg); border-top-style: none;
```

```
                                border-right-style: none; border-
left-style: none; height: 21px; text-align: center;
                                border-bottom-style: none">
                            </td>
                        </tr>
                    </table>
                </td>
                <td colspan="2" rowspan="2">
                <table id="Table3" onclick="return TABLE1_onclick()"
style="border-right: thin solid;
            border-top: thin solid; z-index: 101; left: 1025px;
border-left: thin solid; width: 234px;
            border-bottom: thin solid; position: absolute; top:
175px; height: 172px; background-color: #ffff99">
            <tr>
                <td style="width: 50px; height: 24px">
                    <asp:TextBox ID="TextBox1" runat="server"
BackColor="White" ForeColor="Black" Style="z-index: 100;
                        left: 86px; position: absolute; top: 2px"
Width="137px"></asp:TextBox>
                </td>
                <td style="width: 92px; height: 24px">
                    <asp:TextBox ID="TextBox2" runat="server"
BackColor="White" ForeColor="Black" Style="z-index: 101;
                        left: 85px; position: absolute; top: 36px"
TextMode="Password" Width="136px"></asp:TextBox>
                </td>
            </tr>
            <tr>
                <td style="width: 50px; height: 31px">
                    <asp:Label ID="Label1" runat="server"
BorderColor="Transparent" Font-Bold="True"
                        ForeColor="Black" Style="z-index: 102; left:
6px; position: absolute; top: 7px"
                        Text="Username"></asp:Label>
                </td>
                <td style="width: 92px; height: 31px">
                    <asp:Label ID="Label2" runat="server" Font-
Bold="True" ForeColor="Black" Style="z-index: 103;
                        left: 7px; position: absolute; top: 40px"
Text="Password"></asp:Label>
                </td>
            </tr>
            <tr>
                <td style="width: 50px; height: 27px">
                    <asp:Button ID="Button1" runat="server"
BackColor="Black" Font-Bold="True" ForeColor="#FFFFFF"
                        OnClick="Button1_Click" Style="z-index: 104;
left: 118px; position: absolute;
                        top: 67px; cursor: hand;" Text="Login" />
                </td>
                <td style="width: 92px; height: 27px">
                    <asp:Button ID="Button2" runat="server"
BackColor="Black" Font-Bold="True" ForeColor="White"
                        OnClick="Button2_Click" Style="z-index: 105;
left: 87px; position: absolute;
                        top: 154px; cursor: hand;" Text="Please Sign
Up Here" Width="143px" />
                </td>
            </tr>
            <tr>
```

```
                <td colspan="2" style="height: 23px">
                    <img id="IMG3" onclick="return IMG3_onclick()"
src="NewFolder2/137(1).gif" style="z-index: 115;
                        left: 191px; position: absolute; top:
123px"alt="text" />
                    <asp:Label ID="Label3" runat="server" Font-
Bold="True" Font-Size="Larger" ForeColor="Black"
                        Style="z-index: 109; left: 93px; position:
absolute; top: 126px" Text="New User  ?"
                        Width="93px"></asp:Label>
                </td>
            </tr>
            <tr>
                <td style="width: 50px; height: 56px">
                    <asp:Label ID="Label4" runat="server" Font-
Bold="True" Font-Size="12pt" ForeColor="#660000"
                        Style="z-index: 110; left: 10px; position:
absolute; top: 100px" Text="Label"
                        Visible="False" Width="215px"></asp:Label>
                </td>
                <td style="width: 92px; height: 56px">
                </td>
            </tr>
        </table>
                </td>
            </tr>
            <tr style="font-size: 12pt">
                <td colspan="3" rowspan="4" style="background-image:
url(New Folder3/container.jpg)">
                     
                    <asp:Label ID="Label5" runat="server" Style="z-
index: 100; left: 505px; position: absolute;
                        top: 693px" Text="Label" Font-Bold="True"
Font-Names="Arial" ForeColor="#990000"></asp:Label>
                    <asp:Calendar ID="Calendar1" runat="server"
BackColor="White" BorderColor="Black" Font-Names="Verdana" Font-
Size="9pt"
                        ForeColor="Black" Height="320px" Style="z-
index: 102;
                        left: 1024px; position: absolute; top:
363px" Width="234px" BorderStyle="Solid" CellSpacing="1"
NextPrevFormat="ShortMonth">
                        <SelectedDayStyle BackColor="#333399"
ForeColor="White" />
                        <TodayDayStyle BackColor="#999999"
ForeColor="White" />
                        <OtherMonthDayStyle ForeColor="#999999" />
                        <NextPrevStyle Font-Size="8pt"
ForeColor="White" Font-Bold="True" />
                        <DayHeaderStyle Font-Bold="True"
Height="8pt" Font-Size="8pt" ForeColor="#333333" />
                        <TitleStyle BackColor="#333399" Font-
Bold="True" Font-Size="12pt" ForeColor="White" BorderStyle="Solid"
Height="12pt" />
                        <DayStyle BackColor="#CCCCCC" />
                    </asp:Calendar>
                      

                        
                    <table style="z-index: 106; left: 268px; width:
531px; position: absolute; top: 254px;
```

```
                        height: 419px">
                        <tr>
                            <td colspan="3" style="height: 20px;
text-align: center">
                                <strong><span style="font-size:
16pt; color: #990000">Introduction </span></strong>
                            </td>
                        </tr>
                        <tr>
                            <td colspan="3" style="height: 211px;
text-align: justify">
                                <span style="color: #000099; font-
family: Arial; mso-fareast-font-family: 'Times New Roman';
                                    mso-ansi-language: EN-US; mso-
fareast-language: EN-US; mso-bidi-language: AR-SA">
                                    <span style="font-size: 14pt">
                                        <?xml namespace=""
ns="urn:schemas-microsoft-com:office:smarttags" prefix="st1"
?><st1:country-region><st1:place><SPAN
            style="mso-fareast-font-family: 'Times New Roman'; mso-
ansi-language: EN-US; mso-fareast-language: EN-US; mso-bidi-
language: AR-SA">India</SPAN></st1:place></st1:country-region>
                                          <span style="mso-
fareast-font-family: 'Times New Roman'; mso-ansi-language: EN-US;
                                            mso-fareast-language:
EN-US; mso-bidi-language: AR-SA">with 2.4% of world's geographical
                                            area, is sustaining
about 16% of the global population. Notwithstanding high population
                                            pressure, </span>
                                        <st1:country-
region><st1:place><SPAN
            style="mso-fareast-font-family: 'Times New Roman'; mso-
ansi-language: EN-US; mso-fareast-language: EN-US; mso-bidi-
language: AR-SA">India</SPAN></st1:place></st1:country-region>
                                    </span><span style="color:
#000000; font-family: 'Times New Roman'; mso-fareast-font-family:
'Times New Roman';
                                        mso-ansi-language: EN-US;
mso-fareast-language: EN-US; mso-bidi-language: AR-SA">
                                        <span style="font-size:
14pt"><span style="color: #000099; font-family: Arial">has emerged
                                            from a food deficit to a
self-sufficient status, with three-fold increase in food
                                            grains over a period of
30 years. However, feeding the ever increasing population
                                            would be a challenging
task in future, as the country’s arable land resources are
                                            finite with not much
scope for significantly expanding the area of land under
cultivation.
                                            Therefore, most of the
increase in agricultural production will have to be obtained
                                            through increased
productivity from the existing agricultural land.<br />
                                        </span>     
      <span style="color: #000099; font-family:
Arial">
                                            Evaluation of a strain
for release as a variety consists of various trials and tests
                                            to determine its
superiority over the best existing varieties in yield and other
```

```
                                                    agronomic traits, and
its suitability for consumption.</span></span></span></span></td>
                        </tr>
                    </table>
                    <img src="NewFolder1/rty.gif" style="z-index:
109; left: 49px; width: 132px; position: absolute;
                        top: 327px; height: 103px; font-family:
Times New Roman;" />
                </td>
            </tr>
            <tr style="font-size: 12pt; font-family: Times New
Roman;">
                <td style="width: 57px; height: 114px;">
                </td>
                <td style="height: 114px">
                </td>
            </tr>
            <tr style="font-size: 12pt; font-family: Times New
Roman;">
                <td colspan="2" style="height: 217px; text-align:
center">
                    <strong><span style="color:
#990000"></span></strong></td>
            </tr>
            <tr style="font-size: 12pt; font-family: Times New
Roman;">
                <td colspan="2" style="height: 94px">
                     </td>
            </tr>
            <tr style="font-size: 12pt; font-family: Times New
Roman;">
                <td colspan="6" style="height: 34px; background-
color: #ffffff">
                    <img src="NewFolder3/h2.jpg" style="z-index:
102; left: 265px; position: absolute;
                        top: 243px; width: 531px;"alt="text" />
                </td>
            </tr>
            <tr style="font-size: 12pt; font-family: Times New
Roman;">
                <td colspan="6" style="height: 77px; background-
color: #336633; text-align: center;">
                    <span style="font-size: 10pt"><span style="font-
family: Verdana"><span style="color: #ffcc33">
                        <span style="font-size: 14pt">
                            <br />
                            Indian Agricultural Statistics Research
Institute<br />
                        </span><span style="font-size:
12pt">Division of Computer Application<br />
                            <?xml namespace="" ns="urn:schemas-
microsoft-com:office:smarttags" prefix="st1" ?><?xml namespace=""
prefix="ST1" ?><span style="color: #ffcc33; mso-bidi-font-weight:
bold">Library Avenue</span></span><span
                                style="color: #ffcc33; mso-bidi-
font-weight: bold"><span style="font-size: 12pt;
                                    font-family: Times New Roman">
Pusa, New Delhi-110012 </span><br />
                            </span> </span></span></span></td>
            </tr>
        </table>
```

```
    </form>
</body>
</html>
```

Code for Administrator Login Page

```
<*****************************************************************>

<html xmlns="http://www.w3.org/1999/xhtml" >
<head runat="server">
    <title>Admin Login</title>
</head>
<body >
    <form id="form1" runat="server">

        <asp:Label ID="Label11" runat="server" Font-Size="XX-Large"
ForeColor="Blue" Style="z-index: 100;
            left: 604px; position: absolute; top: 520px"
Text="Label" Width="318px"></asp:Label>
        <asp:Panel ID="Panel1" runat="server" Height="61px"
Style="z-index: 101; left: 554px;
            position: absolute; top: 507px" Width="347px">
        </asp:Panel>
         
          
         
        <table style="z-index: 103; left: 337px; width: 583px;
position: absolute; top: 139px; height: 398px;">
            <tr>
                <td colspan="3">
                    <asp:Label ID="Label1" runat="server"
ForeColor="#003300" Style="z-index: 100; left: 6px;
            position: absolute; top: 46px" Text="First Name" Font-
Bold="True"></asp:Label>
                <asp:TextBox ID="TextBox2" runat="server" Style="z-
index: 113; left: 197px; position: absolute;
            top: 81px"></asp:TextBox>
                <asp:RequiredFieldValidator
ID="RequiredFieldValidator1" runat="server"
ControlToValidate="TextBox1"
            ErrorMessage="RequiredFieldValidator" Style="z-index:
120; left: 395px; position: absolute;
            top: 45px" Width="152px" Font-Bold="True">First name is
required</asp:RequiredFieldValidator>
                </td>
            </tr>
            <tr>
                <td><asp:Label ID="Label2" runat="server"
ForeColor="#003300" Style="z-index: 101; left: 3px;
            position: absolute; top: 82px" Text="Lat Name" Font-
Bold="True"></asp:Label>
                </td>
                <td><asp:TextBox ID="TextBox1" runat="server"
Style="z-index: 112; left: 197px; position: absolute;
            top: 44px"></asp:TextBox>
                </td>
                <td><asp:RequiredFieldValidator
ID="RequiredFieldValidator2" runat="server"
ControlToValidate="TextBox2"
            ErrorMessage="RequiredFieldValidator" Style="z-index:
121; left: 399px; position: absolute;
```

```
            top: 79px" Width="152px" Font-Bold="True">Last name is
required</asp:RequiredFieldValidator>
                </td>
            </tr>
            <tr>
                <td><asp:Label ID="Label3" runat="server"
ForeColor="#003300" Style="z-index: 102; left: 9px;
            position: absolute; top: 115px" Text="Login Id" Font-
Bold="True"></asp:Label>
                </td>
                <td><asp:TextBox ID="TextBox5" runat="server"
Style="z-index: 116; left: 198px; position: absolute;
            top: 190px" TextMode="Password"></asp:TextBox>
                </td>
                <td><asp:RequiredFieldValidator
ID="RequiredFieldValidator3" runat="server"
ControlToValidate="TextBox3"
            ErrorMessage="RequiredFieldValidator" Style="z-index:
122; left: 399px; position: absolute;
            top: 118px" Width="137px" Font-Bold="True">Login id is
required</asp:RequiredFieldValidator>
                </td>
            </tr>
            <tr>
                <td><asp:Label ID="Label4" runat="server"
ForeColor="#003300" Style="z-index: 103; left: 3px;
            position: absolute; top: 153px" Text="Password" Font-
Bold="True"></asp:Label>
                </td>
                <td><asp:TextBox ID="TextBox3" runat="server"
Style="z-index: 114; left: 198px; position: absolute;
            top: 120px"></asp:TextBox>
                </td>
                <td><asp:RequiredFieldValidator
ID="RequiredFieldValidator4" runat="server"
ControlToValidate="TextBox4"
            ErrorMessage="RequiredFieldValidator" Style="z-index:
123; left: 396px; position: absolute;
            top: 153px" Width="143px" Font-Bold="True">Password is
required</asp:RequiredFieldValidator>
                </td>
            </tr>
            <tr>
                <td><asp:Label ID="Label5" runat="server"
ForeColor="#003300" Height="18px" Style="z-index: 104;
            left: 0px; position: absolute; top: 191px" Text="Confirm
Password" Font-Bold="True"></asp:Label>
                </td>
                <td><asp:TextBox ID="TextBox4" runat="server"
Style="z-index: 115; left: 198px; position: absolute;
            top: 154px" TextMode="Password"></asp:TextBox>
                </td>
                <td><asp:CompareValidator ID="CompareValidator1"
runat="server" ControlToCompare="TextBox5"
            ControlToValidate="TextBox4"
ErrorMessage="CompareValidator" Style="z-index: 124;
            left: 397px; position: absolute; top: 189px"
Width="172px" Font-Bold="True">Password should be
same</asp:CompareValidator>
                </td>
            </tr>
```

```
            <tr>
                <td><asp:Label ID="Label6" runat="server"
ForeColor="#003300" Style="z-index: 105; left: 7px;
            position: absolute; top: 226px" Text="Designation" Font-
Bold="True"></asp:Label>
                </td>
                <td><asp:TextBox ID="TextBox6" runat="server"
Style="z-index: 117; left: 198px; position: absolute;
            top: 226px"></asp:TextBox>
                </td>
                <td><asp:RequiredFieldValidator
ID="RequiredFieldValidator5" runat="server"
ControlToValidate="TextBox6"
            ErrorMessage="RequiredFieldValidator" Style="z-index:
125; left: 394px; position: absolute;
            top: 230px" Width="155px" Font-Bold="True">Designation
is required</asp:RequiredFieldValidator>
                </td>
            </tr>
            <tr>
                <td><asp:Label ID="Label7" runat="server"
ForeColor="#003300" Style="z-index: 106; left: 6px;
            position: absolute; top: 261px" Text="Department" Font-
Bold="True"></asp:Label>
                </td>
                <td><asp:TextBox ID="TextBox7" runat="server"
Style="z-index: 118; left: 199px; position: absolute;
            top: 263px"></asp:TextBox>
                </td>
                <td><asp:RequiredFieldValidator
ID="RequiredFieldValidator6" runat="server"
ControlToValidate="TextBox7"
            ErrorMessage="RequiredFieldValidator" Style="z-index:
126; left: 397px; position: absolute;
            top: 262px" Width="158px" Font-Bold="True">Department is
required</asp:RequiredFieldValidator>
                </td>
            </tr>
            <tr>
                <td><asp:Label ID="Label8" runat="server"
ForeColor="#003300" Style="z-index: 107; left: 4px;
            position: absolute; top: 295px" Text="Organization"
Font-Bold="True"></asp:Label>
                </td>
                <td><asp:TextBox ID="TextBox8" runat="server"
Style="z-index: 119; left: 198px; position: absolute;
            top: 297px"></asp:TextBox>
                </td>
                <td><asp:RequiredFieldValidator
ID="RequiredFieldValidator7" runat="server"
ControlToValidate="TextBox8"
            ErrorMessage="RequiredFieldValidator" Style="z-index:
127; left: 396px; position: absolute;
            top: 300px" Width="172px" Font-Bold="True">Organization
is required</asp:RequiredFieldValidator>
                </td>
            </tr>
            <tr>
                <td><asp:Label ID="Label9" runat="server"
ForeColor="#003300" Style="z-index: 108; left: 10px;
```

```
            position: absolute; top: 332px" Font-
Bold="True">Check</asp:Label>
                </td>
                <td><asp:DropDownList ID="DropDownList1"
runat="server" Style="z-index: 128; left: 199px;
            position: absolute; top: 335px" Width="158px">
            <asp:ListItem>admin</asp:ListItem>
            <asp:ListItem>power</asp:ListItem>
        </asp:DropDownList>
                </td>
                <td><asp:Button ID="Button1" runat="server" Font-
Bold="True" ForeColor="#003300" OnClick="Button1_Click"
            Style="z-index: 129; left: 3px; position: absolute; top:
368px" Text="Register" />
                </td>
            </tr>
            <tr>
                <td><asp:Label ID="Label10" runat="server" Font-
Bold="True" Font-Names="Monotype Corsiva"
            Font-Size="X-Large" ForeColor="#FFFF00" Height="32px"
Style="z-index: 109; left: 185px;
            position: absolute; top: 2px" Text="Please     signup
here!" Width="205px" BackColor="#006600"></asp:Label>
                </td>
                <td><asp:Button ID="Button2" runat="server" Font-
Bold="True" ForeColor="#003300" OnClick="Button2_Click"
            Style="z-index: 130; left: 198px; position: absolute;
top: 370px" Text="Reset" />
                </td>
                <td>
                </td>
            </tr>
            <tr>
                <td>
                </td>
                <td colspan="2">
                </td>
            </tr>
        </table>
        <img src="NewFolder2/8.gif" style="z-index: 104; left: 0px;
position: absolute; top: 0px"
            width="1257" />
    </form>
</body>
</html>
```

Code for Administrator Login Page

```
<*********************************************************************>

using System;
using System.Data;
using System.Configuration;
using System.Collections;
using System.Web;
using System.Web.Security;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Web.UI.HtmlControls;
using System.Data.SqlClient;

public partial class Admin : System.Web.UI.Page
```

```
{
    protected void Page_Load(object sender, EventArgs e)
    {
        Label11.Visible = false;
        Panel1.Visible = false;
        Page.Header.Title = "Information System For Varietal
Experiments";
    }

    protected void Button1_Click(object sender, EventArgs e)
    {
        try
        {
            SqlConnection con = new SqlConnection();
            con.ConnectionString = "Data Source =localhost;Initial
Catalog=isve;integrated security=true;";

            con.Open();

            SqlDataAdapter da = new SqlDataAdapter("select * from
usr", con);

            SqlCommandBuilder cb = new SqlCommandBuilder(da);

            DataSet ds = new DataSet("usr");

            da.Fill(ds, "usr");

            DataRow row = ds.Tables["usr"].NewRow();

            string fname, lname, id, pas, desig, deprt, org,chk;


            fname = TextBox1.Text;
            lname = TextBox2.Text;
            id = TextBox3.Text;
            pas = TextBox4.Text;
            desig = TextBox6.Text;
            deprt = TextBox7.Text;
            org = TextBox8.Text;
            chk = DropDownList1.Text;

            row["firstname"] = fname;
            row["lastname"] = lname;
            row["login_id"] = id;
            row["password"] = pas;
            row["designation"] = desig;
            row["department"] = deprt;
            row["organization"] = org;
            row["chk"] = chk;



            ds.Tables["usr"].Rows.Add(row);

            da.Update(ds, "usr");
            con.Close();
            Cache.Insert("txt", TextBox1.Text);
            Server.Transfer("title_page.aspx");

        }
```

```
        catch (Exception rt)
        {
            Label11.Visible = true;
            Panel1.Visible = true;
            Label11.Text = "Choose another Login";
            TextBox3.Text = "";
        }

    }
    protected void Button2_Click(object sender, EventArgs e)
    {
        TextBox1.Text = "";
        TextBox2.Text = "";
        TextBox3.Text = "";
        TextBox4.Text = "";
        TextBox5.Text = "";
        TextBox6.Text = "";
        TextBox7.Text = "";
        TextBox8.Text = "";
    }
}
```

Code (HTML) for Data Management Page

```
<**********************************************************************>
<html xmlns="http://www.w3.org/1999/xhtml" >
<head runat="server">
    <title>Data management</title>
</head>
<body text="#990099">
    <form id="form1" runat="server">
    <div style="font-size: 20pt; vertical-align: middle; color:
#ff1493; line-height: normal; font-style: italic; font-family:
'Monotype Corsiva', Monospace; letter-spacing: normal; text-align:
left" title="data management">
         

        <br />
        <span style="font-family: Arial Black">


            <br />

                  <br />


             <br />
            <br />
                           
       
              
            <table style="z-index: 104; left: 0px; width: 1259px;
font-style: italic; position: absolute;
                top: 0px; height: 257px">
                <tr>
                    <td colspan="3" style="height: 128px">
                        <img src="NewFolder2/8.gif" style="z-index:
103; left: 0px; position: absolute; top: 0px"
                            width="1257"alt="text" />
                    </td>
                </tr>
```

```
                <tr>
                    <td style="height: 60px; width: 397px;">
                         </td>
                    <td style="height: 60px; width: 376px; text-
align: center;">
                        <span style="font-size: 24pt; color:
#009900; font-family: Times New Roman; text-decoration: underline">
                            <strong>Data
Management</strong></span></td>
                    <td style="height: 60px">
                    </td>
                </tr>
                <tr>
                    <td style="height: 367px; width: 397px;"><table
style="z-index: 102; left: 456px; position: absolute; top: 221px;
width: 312px; height: 276px; font-style: italic;">
                        <tr>
                            <td style="width: 38px"><img
src="NewFolder2/button-01.gif" style="z-index: 109; left: 9px;
width: 27px;
                        position: absolute; top: 9px; height: 27px"
alt="text" />
                            </td>
                            <td><asp:HyperLink ID="HyperLink1"
runat="server" Font-Bold="True" Font-Names="Monotype Corsiva"
                ForeColor="#003300" NavigateUrl="~/modifyZone.aspx"
Style="z-index: 104; left: 48px;
                position: absolute; top: 6px">Zone</asp:HyperLink>
                            </td>

                        </tr>
                        <tr>
                            <td style="width: 38px"><img
src="NewFolder2/button-01.gif" style="z-index: 115; left: 9px;
width: 27px;
                        position: absolute; top: 50px; height: 27px"
alt="text"/>
                            </td>
                            <td><asp:HyperLink ID="HyperLink2"
runat="server" Font-Bold="True" Font-Names="Monotype Corsiva"
                ForeColor="#003300" Style="z-index: 105; left: 49px;
position: absolute; top: 48px"
NavigateUrl="~/modifystate.aspx">State</asp:HyperLink>
                            </td>
                        </tr>
                        <tr>
                            <td style="width: 38px; height:
17px;"><img src="NewFolder2/button-01.gif" style="z-index: 110;
left: 11px; width: 27px;
                        position: absolute; top: 259px; height:
27px" alt="text"/>
                            </td>
                            <td style="height: 17px"><asp:HyperLink
ID="HyperLink3" runat="server" Font-Bold="True" Font-Names="Monotype
Corsiva"
                        ForeColor="#003300" Style="z-index: 107;
left: 47px; position: absolute; top: 89px"
NavigateUrl="~/ModifyCentre.aspx">Centre</asp:HyperLink>
                            </td>

                        </tr>
```

```
                        <tr>
                            <td style="width: 38px"><img
src="NewFolder2/button-01.gif" style="z-index: 111; left: 8px;
width: 27px;
                        position: absolute; top: 218px; height:
27px" alt="text"/>
                            </td>
                            <td><asp:HyperLink ID="HyperLink4"
runat="server" Font-Bold="True" Font-Names="Monotype Corsiva"
                        ForeColor="#003300" Style="z-index: 108;
left: 49px; position: absolute; top: 134px"
NavigateUrl="~/modifyvariety.aspx">Variety</asp:HyperLink>
                            </td>

                        </tr>
                        <tr>
                            <td style="width: 38px"><img
src="NewFolder2/button-01.gif" style="z-index: 112; left: 8px;
width: 27px;
                        position: absolute; top: 178px; height:
27px" alt="text"/>
                            </td>
                            <td><asp:HyperLink ID="HyperLink5"
runat="server" Font-Bold="True" Font-Names="Monotype Corsiva"
            ForeColor="#003300" Style="z-index: 100; left: 48px;
position: absolute; top: 176px"
NavigateUrl="~/modifytrialtype.aspx">Trial Type</asp:HyperLink>
                            </td>

                        </tr>
                        <tr>
                            <td style="width: 38px"><img
src="NewFolder2/button-01.gif" style="z-index: 113; left: 9px;
width: 27px;
                        position: absolute; top: 134px; height:
27px" alt="text"/>
                            </td>
                            <td><asp:HyperLink ID="HyperLink6"
runat="server" Font-Bold="True" Font-Names="Monotype Corsiva"
                ForeColor="#003300" Style="z-index: 101; left: 49px;
position: absolute; top: 215px"
NavigateUrl="~/modifytrialseries.aspx">Trial Series</asp:HyperLink>
                            </td>

                        </tr>
                        <tr>
                            <td style="width: 38px; height:
32px;"><img src="NewFolder2/button-01.gif" style="z-index: 114;
left: 9px; width: 27px;
                        position: absolute; top: 90px; height: 27px"
alt="text"/>
                            </td>
                            <td style="height: 32px"><asp:HyperLink
ID="HyperLink7" runat="server" Font-Bold="True" Font-Names="Monotype
Corsiva"
                ForeColor="#003300" Style="z-index: 102; left: 50px;
position: absolute; top: 256px" Width="226px"
NavigateUrl="~/modifycontrolvariety.aspx">Variety For
Control</asp:HyperLink>
                            </td>
```

```
                        </tr>
                    </table>
                    </td>
                    <td style="height: 367px; width: 376px;">
                    </td>
                    <td style="height: 367px">
                    </td>
                </tr>
                <tr>
                    <td style="height: 163px; width: 397px;">
                    </td>
                    <td style="height: 163px; width: 376px;">
                    </td>
                    <td style="height: 163px">
                    </td>
                </tr>
                <tr>
                    <td style="width: 397px">
                    </td>
                    <td style="width: 376px">
                    </td>
                    <td>
                    </td>
                </tr>
                <tr>
                    <td colspan="3" style="height: 90px; background-
color: #336633">
                        <span style="font-size: 10pt; font-family:
Verdana"></span>
                    </td>
                </tr>
            </table>
            <span style="color: #ffff00;
                font-family: Monotype Corsiva"><strong><br />
                                 
         
                        

              

                    <br />
                                 
        

                             
  <br />
                                 
         
                            

                     <br />
                                 
         
                             <br />
                                 
        
                    <br />
                                 
        
                    </strong></span></span>
        <br />
```

```
        <br />
        <br />
        <br />
         </div>
    </form>
</body>
</html>
```

Code for Zone Management page (Addition)

```
<*****************************************************************>

using System;
using System.Data;
using System.Configuration;
using System.Collections;
using System.Web;
using System.Web.Security;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Web.UI.HtmlControls;
using System.Data.SqlClient;

public partial class DefaultZone_add : System.Web.UI.Page
{




    protected void Button1_Click1(object sender, EventArgs e)
    {
        if (TextBox1.Text == "" && TextBox2.Text == "")
        {
            Label3.Visible = true;
            Label3.Text = "Please enter zone name and abbreviation
!";
        }
        else if (TextBox1.Text == "")
        {
            Label3.Visible = true;
            Label3.Text = "Please enter zone name !";
        }
        else if (TextBox2.Text == "")
        {
            Label3.Visible = true;
            Label3.Text = "Please enter Abbreviation !";
        }
        else
        {

        string code = TextBox1.Text;
        string constring = "Data source = localhost; initial
catalog=ISVE; integrated security=true;";
        string concommand = "select * from zone where
zone_name=@empname";
        SqlConnection con1 = new SqlConnection(constring);
        con1.Open();

        SqlCommand cmd = new SqlCommand(concommand, con1);
        cmd.Parameters.Add(new SqlParameter("@empname", code));
        SqlDataReader dr = cmd.ExecuteReader();
```

```
        while (dr.Read())
        {
            HiddenField1.Value = dr[1].ToString();
            HiddenField2.Value = dr[2].ToString();
        }
        if (TextBox1.Text == HiddenField1.Value && TextBox2.Text ==
HiddenField2.Value)
        {
            Label3.Text = "This zone is allready exist !";
        }
        else
        {

          try
            {
                SqlConnection con = new SqlConnection();
                con.ConnectionString = "Data Source
=localhost;Initial Catalog=isve;integrated security=true;";

                con.Open();

                SqlDataAdapter da = new SqlDataAdapter("select *
from Zone", con);

                SqlCommandBuilder cb = new SqlCommandBuilder(da);

                DataSet ds = new DataSet("Zone");

                da.Fill(ds, "Zone");

                DataRow row = ds.Tables["Zone"].NewRow();

                string zname, abbre;


                zname = TextBox1.Text;
                abbre = TextBox2.Text;

                row["Zone_name"] = zname;
                row["Abbreviation"] = abbre;

                ds.Tables["Zone"].Rows.Add(row);

                da.Update(ds, "Zone");
                con.Close();
                Label3.Visible = true;
                Panel1.Visible = true;
                Label3.Text = "New Zone is added successfully !";

            }
            catch (Exception rt)
            {
                Label3.Visible = true;
                Panel1.Visible = true;
                Label3.Text = "This Zone is already exist !";
                TextBox1.Text = "";
                TextBox2.Text = "";
            }
            }
        }
    }
```

```
    protected void Button2_Click(object sender, EventArgs e)
    {
        TextBox1.Text = "";
        TextBox2.Text = "";
    }
    protected void Page_Load(object sender, EventArgs e)
    {
        Label3.Visible = false;
        Page.Header.Title = "Information System For Varietal
Experiments";
    }
    protected void Button3_Click(object sender, EventArgs e)
    {
        Server.Transfer("home.aspx");
    }
    protected void Button5_Click(object sender, EventArgs e)
    {
        Server.Transfer("modifyzone.aspx");
    }
    protected void Button6_Click(object sender, EventArgs e)
    {
        Server.Transfer("modifyzone.aspx");
    }
    protected void Button8_Click(object sender, EventArgs e)
    {
        Server.Transfer("titlepage.aspx");
    }
}
```

Code for Zone Management page (Modification, Deletion)

```
<*****************************************************************>

using System;
using System.Data;
using System.Configuration;
using System.Collections;
using System.Web;
using System.Web.Security;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Web.UI.HtmlControls;
using System.Data.SqlClient;
public partial class DeleteZone : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        Label4.Visible = false;
        string constring = "Data source = localhost; initial
catalog=ISVE; integrated security=true;";
        SqlConnection con = new SqlConnection(constring);
        con.Open();

        SqlDataAdapter da = new SqlDataAdapter("select zone_id as
'Zone ID', Zone_Name as 'Zone Name',abbreviation as 'Abbreviation'
from Zone", con);
        DataSet ds = new DataSet("zone");
        da.Fill(ds, "zone");

        GridView1.DataSource = ds;
        GridView1.DataBind();
```

```
        con.Close();

        Button1.Attributes.Add("onclick", "return
confirm_delete();");

        Button2.Attributes.Add("onclick", "return confirm_update
();");

    }

    protected void GridView1_SelectedIndexChanged(object sender,
EventArgs e)
    {

    }

    protected void GridView1_SelectedIndexChanged1(object sender,
EventArgs e)
    {
        Button1.Enabled = true;
        Button2.Enabled = true;
        Label3.Visible = false;
        Label4.Visible = false;
        HiddenField1.Value = GridView1.SelectedRow.Cells[1].Text;
        TextBox1.Text = GridView1.SelectedRow.Cells[2].Text;
        TextBox2.Text = GridView1.SelectedRow.Cells[3].Text;

    }
    protected void TextBox2_TextChanged(object sender, EventArgs e)
    {

    }
    protected void Button1_Click(object sender, EventArgs e)
    {
        if (TextBox1.Text == "" || TextBox2.Text == "")
        {
            Label4.Visible = true;
            Label3.Visible = false;
            Label4.Text = "Please select record which you want to
Delete !";
        }
        else
        {

            string code = HiddenField1.Value;
            string constring = "Data source = localhost; initial
catalog=ISVE; integrated security=true;";
            string concommand = "delete from zone where
zone_id=@empcode";
            SqlConnection con = new SqlConnection(constring);
            con.Open();
            SqlCommand cmd = new SqlCommand(concommand, con);
            cmd.Parameters.Add(new SqlParameter("@empcode", code));
            int r = cmd.ExecuteNonQuery();
            if (r >= 1)
            {
                Label3.Visible = true;
                Label3.Text = "Zone has been deleted !";
            }
            else
            {
```

```
                Label3.Visible = true;
                Label3.Text = "Wrong work";
            }

            con.Close();
            TextBox1.Text = "";
            TextBox2.Text = "";
            this.Page_Load(null, null);
            Label4.Visible = false;
        }
        Button1.Enabled = false;
        Button2.Enabled = false;
    }
    protected void Button2_Click(object sender, EventArgs e)
    {
        if (TextBox1.Text == "" || TextBox2.Text == "")
        {
            Label4.Visible = true;
            Label4.Text = "Please select record which you want to
Update !";
        }
        else
        {

            string name = TextBox1.Text;
            string abrre = TextBox2.Text;
            string zoneid = HiddenField1.Value;


            string constring = "Data source = localhost; initial
catalog=ISVE; integrated security=true;";
            string concommand = "update zone set
zone_name=@empname,abbreviation=@empaddress where zone_id=@empcode";
            SqlConnection con = new SqlConnection(constring);
            con.Open();
            SqlCommand cmd = new SqlCommand(concommand, con);
            cmd.Parameters.Add(new SqlParameter("@empcode",
zoneid));
            cmd.Parameters.Add(new SqlParameter("@empname", name));
            cmd.Parameters.Add(new SqlParameter("@empaddress",
abrre));

            int u = cmd.ExecuteNonQuery();
            if (u >= 1)
            {
                Label3.Visible = true;
                Label3.Text = "Zone has been updated !";
            }
            con.Close();
            TextBox1.Text = "";
            TextBox2.Text = "";
            this.Page_Load(null, null);
            Label4.Visible = false;
        }
        Button1.Enabled = false;
        Button2.Enabled = false;
    }
    protected void Button3_Click(object sender, EventArgs e)
    {
        Server.Transfer("home.aspx");
    }
```

```
    protected void Button4_Click(object sender, EventArgs e)
    {
        Server.Transfer("addzone.aspx");
    }
    protected void Button8_Click(object sender, EventArgs e)
    {
        Server.Transfer("default.aspx");
    }
    protected void Button7_Click(object sender, EventArgs e)
    {
        Server.Transfer("help.aspx");
    }
    protected void GridView1_RowDataBound(object sender,
GridViewRowEventArgs e)
    {
        if (e.Row.RowType == DataControlRowType.DataRow)
        {
            e.Row.Cells[1].Visible = false;

        }
        if (e.Row.RowType == DataControlRowType.Header)
        {
            e.Row.Cells[1].Visible = false;

        }
    }
}
```

Code for Varietal Trial Report

```
<*********************************************************************>

using System;
using System.Data;
using System.Configuration;
using System.Collections;
using System.Web;
using System.Web.Security;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Web.UI.HtmlControls;
using System.Data.SqlClient;

public partial class Report2 : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {

        if (IsPostBack == false  )
        {
            Panel1.Visible = false;
            Panel2.Visible = false;
            Panel3.Visible = false;
            string constring1 = "Data source = localhost; initial
catalog=ISVE; integrated security=true;";
            string concommand1 = "SELECT  distinct
Trial_series.Trial_series_name FROM Varietal_trial INNER JOIN
Trial_series ON Varietal_trial.Trial_series_id =
Trial_series.Trial_series_id";
            SqlConnection con1 = new SqlConnection(constring1);
            con1.Open();
            SqlCommand cmd1 = new SqlCommand(concommand1, con1);
```

```
            SqlDataReader dr1 = cmd1.ExecuteReader();
            while (dr1.Read())
            {
                DropDownList1.Items.Add(dr1[0].ToString());
            }
        }
        FillDesign();
        FillCharacter();
        FillTrialType();
        FillYear();
        FillTrialSeries();
    }
    protected void DropDownList1_SelectedIndexChanged(object sender,
EventArgs e)
    {


    }
    private void FillCharacter()
    {
        if (IsPostBack == false)
        {
            string constring1 = "Data source = localhost; initial
catalog=ISVE; integrated security=true;";
            string concommand1 = "select distinct Charachter from
DataEntry";
            SqlConnection con1 = new SqlConnection(constring1);
            con1.Open();
            SqlCommand cmd1 = new SqlCommand(concommand1, con1);
            SqlDataReader dr1 = cmd1.ExecuteReader();
            while (dr1.Read())
            {
                DropDownList3.Items.Add(dr1[0].ToString());
            }
        }
    }
    private void FillDesign()
    {
        if (IsPostBack == false)
        {
            string constring1 = "Data source = localhost; initial
catalog=ISVE; integrated security=true;";
            string concommand1 = "SELECT distinct Design.Design_name
FROM Varietal_trial INNER JOIN Design ON Varietal_trial.Design_id =
Design.Design_id INNER JOIN Trial_series ON
Varietal_trial.Trial_series_id = Trial_series.Trial_series_id";
            SqlConnection con1 = new SqlConnection(constring1);
            con1.Open();
            SqlCommand cmd1 = new SqlCommand(concommand1, con1);
            SqlDataReader dr1 = cmd1.ExecuteReader();
            while (dr1.Read())
            {
                DropDownList2.Items.Add(dr1[0].ToString());
            }
        }
    }
    protected void Button1_Click(object sender, EventArgs e)
    {

        //BindData();
    }
```

```
    public void BindData()
    {
        SqlConnection myConnection = new
SqlConnection("Server=.;database=ISVE;trusted_connection=true");

SqlDataAdapter ad = new SqlDataAdapter("SELECT  DataEntry.V1,
DataEntry.V2, DataEntry.V3, DataEntry.V4, DataEntry.V5,DataEntry.V6,
DataEntry.V7, DataEntry.V8, DataEntry.V9, DataEntry.V10,
DataEntry.V11,DataEntry.V12, DataEntry.V13, DataEntry.V14,
DataEntry.V15 FROM DataEntry INNER JOIN Varietal_trial ON
DataEntry.Varietal_trial_Id = Varietal_trial.Varietal_trial_id INNER
JOIN Trial_series ON Varietal_trial.Trial_series_id =
Trial_series.Trial_series_id INNER JOIN Design ON
Varietal_trial.Design_id = Design.Design_id WHERE
Trial_series.Trial_series_name='" + this.DropDownList1.Text + "' and
Design.Design_name='" + this.DropDownList2.Text + "' and
DataEntry.Charachter='" + this.DropDownList3.Text + "'",
myConnection);

        DataSet ds = new DataSet();

        ad.Fill(ds, "DataEntry");

        this.DataGrid1.DataSource = ds;

        this.DataGrid1.DataBind();

    }
    protected void Button2_Click(object sender, EventArgs e)
    {
        this.DataGrid1.DataSource = null;
        this.DataGrid1.DataBind();
        SqlConnection myConnection = new
SqlConnection("Server=.;database=ISVE;trusted_connection=true");

SqlDataAdapter ad = new SqlDataAdapter("SELECT  distinct
State.State_name, centre.Center_Name, Variety.Variety_name FROM
centre INNER JOIN Var_trial_centre_variety INNER JOIN Varietal_trial
ON Var_trial_centre_variety.Varital_trial_id =
Varietal_trial.Varietal_trial_id INNER JOIN Variety ON
Var_trial_centre_variety.Variety_id = Variety.Variety_id INNER JOIN
Trial_series ON Varietal_trial.Trial_series_id =
Trial_series.Trial_series_id ON centre.Centre_id = Variety.Centre_id
INNER JOIN State ON centre.state_id = State.State_id where
Trial_series.Trial_series_name='" + this.DropDownList4.Text + "'",
myConnection);

        DataSet ds = new DataSet();

        ad.Fill(ds, "DataEntry");

        this.DataGrid1.DataSource = ds;

        this.DataGrid1.DataBind();
    }
    private void FillTrialType()
    {
        if (IsPostBack == false)
        {
            string constring1 = "Data source = localhost; initial
catalog=ISVE; integrated security=true;";
```

```
            string concommand1 = "SELECT  distinct
Trial_type.Trial_type_name FROM Varietal_trial INNER JOIN
Trial_series ON Varietal_trial.Trial_series_id =
Trial_series.Trial_series_id INNER JOIN Trial_type ON
Trial_series.Trial_type_id = Trial_type.Trial_type_id ";
            SqlConnection con1 = new SqlConnection(constring1);
            con1.Open();
            SqlCommand cmd1 = new SqlCommand(concommand1, con1);
            SqlDataReader dr1 = cmd1.ExecuteReader();
            while (dr1.Read())
            {
                this.dRTrialType.Items.Add(dr1[0].ToString());
            }
        }
    }
    private void FillYear()
    {
        if (IsPostBack == false)
        {
            string constring1 = "Data source = localhost; initial
catalog=ISVE; integrated security=true;";
            string concommand1 = "select distinct [year] from
varietal_trial order by [year]";
            SqlConnection con1 = new SqlConnection(constring1);
            con1.Open();
            SqlCommand cmd1 = new SqlCommand(concommand1, con1);
            SqlDataReader dr1 = cmd1.ExecuteReader();
            while (dr1.Read())
            {
                this.DrYear.Items.Add(dr1[0].ToString());
            }
        }
    }
    protected void Button3_Click(object sender, EventArgs e)
    {
        SqlConnection myConnection = new
SqlConnection("Server=.;database=ISVE;trusted_connection=true");

        SqlDataAdapter ad = new SqlDataAdapter("SELECT
Trial_series.Trial_series_name AS Trial_series_name,
[Zone].Zone_name AS Zone_name, Variety.Variety_name AS Variety_name
FROM Variety INNER JOIN Var_trial_centre_variety ON
Variety.Variety_id = Var_trial_centre_variety.Variety_id INNER JOIN
Varietal_trial INNER JOIN Trial_series ON
Varietal_trial.Trial_series_id = Trial_series.Trial_series_id INNER
JOIN [Zone] ON Varietal_trial.zone_id = [Zone].Zone_id ON
Var_trial_centre_variety.Varital_trial_id =
Varietal_trial.Varietal_trial_id INNER JOIN Trial_type ON
Trial_series.Trial_type_id = Trial_type.Trial_type_id WHERE
Trial_type.Trial_type_name='" + this.dRTrialType.Text + "' AND
Varietal_trial.YEAR='" + this.DrYear.Text + "'", myConnection);

        DataSet ds = new DataSet();

        ad.Fill(ds, "DataEntry");

        this.DataGrid1.DataSource = ds;

        this.DataGrid1.DataBind();
    }
    private void FillTrialSeries()
```

```
    {
        if (IsPostBack == false)
        {
            string constring1 = "Data source = localhost; initial
catalog=ISVE; integrated security=true;";
            string concommand1 = "SELECT  distinct
Trial_series.Trial_series_name FROM Varietal_trial INNER JOIN
Trial_series ON Varietal_trial.Trial_series_id =
Trial_series.Trial_series_id";
            SqlConnection con1 = new SqlConnection(constring1);
            con1.Open();
            SqlCommand cmd1 = new SqlCommand(concommand1, con1);
            SqlDataReader dr1 = cmd1.ExecuteReader();
            while (dr1.Read())
            {
                DropDownList4.Items.Add(dr1[0].ToString());
            }
        }

    }
    protected void RadioButton1_CheckedChanged(object sender,
EventArgs e)
    {
        if (IsPostBack == true )
        {
            this.Panel1.Visible = true;
            this.Panel2.Visible = false;
            this.Panel3.Visible = false;
        }

    }
    protected void RadioButton2_CheckedChanged(object sender,
EventArgs e)
    {
        if (IsPostBack == true)
        {
            this.Panel1.Visible = false ;
            this.Panel2.Visible = true ;
            this.Panel3.Visible = false;
        }
    }
    protected void RadioButton3_CheckedChanged(object sender,
EventArgs e)
    {
        if (IsPostBack == true)
        {
            this.Panel1.Visible = false ;
            this.Panel2.Visible = false;
            this.Panel3.Visible = true ;
        }
    }
```

Code for Experimental Detail Report (Detail Page)

```
<*****************************************************************>

Imports System.Data.SqlClient
Imports System.Data
Imports CrystalDecisions.Shared
Imports CrystalDecisions.CrystalReports.Engine
Imports System.Web.UI.Page
Imports CrystalDecisions.Web.PrintMode
Imports System.Drawing.Printing
```

```
Partial Class NewReport
    Inherits System.Web.UI.Page

    Protected Sub Button1_Click(ByVal sender As Object, ByVal e As
System.EventArgs) Handles Button1.Click
        SearchVID()
        Dim str1, str2, str3, str4 As String
        str1 = "'" & Me.DropDownList1.Text & "'"
        str2 = "'" & Me.DropDownList2.Text & "'"
        str3 = "'" & Me.DropDownList3.Text & "'"
        str4 = "'" & Me.DropDownList4.Text & "'"

        Dim con As New SqlConnection
        con.ConnectionString =
"Server=.;database=ISVE;trusted_connection=true"
        Dim da As New SqlDataAdapter("SELECT
Trial_type.Trial_type_name AS Trial_type_name,
Trial_series.Trial_series_name AS Trial_series_name,
Varietal_trial.[Year] AS Year, [Zone].Zone_name AS Zone_name,
State.State_name AS State_name, Design.Design_name AS Design_name,
centre.Center_Name AS Center_Name, Var_trial_centre.Replication_no
AS Replication_no, Var_trial_centre.Date_of_sowingfrom AS
Date_of_sowingfrom, Var_trial_centre.Date_of_sowingto AS
Date_of_sowingto, Var_trial_centre.Plotsize_gross_length AS
Plotsize_gross_length, Var_trial_centre.Seed_requirements as
Seed_requirements, Var_trial_centre.Plotsize_gross_breadth AS
Plotsize_gross_breadth, Var_trial_centre.Plotsize_Net_length AS
Plotsize_Net_length, Var_trial_centre.Plotsize_Net_breadth AS
Plotsize_Net_breadth, Var_trial_centre.Fertilizerdose_N AS
Fertilizerdose_N, Var_trial_centre.Fertilizerdose_K AS
Fertilizerdose_K, Var_trial_centre.Fertilizerdose_P AS
Fertilizerdose_P,Var_trial_centre.Seed_rate AS Seed_rate FROM centre
INNER JOIN State INNER JOIN Trial_type INNER JOIN Trial_series ON
Trial_type.Trial_type_id = Trial_series.Trial_type_id INNER JOIN
Varietal_trial ON Trial_series.Trial_series_id =
Varietal_trial.Trial_series_id INNER JOIN Var_trial_centre ON
Varietal_trial.Varietal_trial_id =
Var_trial_centre.Varietal_trial_id INNER JOIN [Zone] ON
Varietal_trial.zone_id = [Zone].Zone_id ON State.State_id =
[Zone].State_id INNER JOIN Design ON Varietal_trial.Design_id =
Design.Design_id ON centre.Centre_id = Var_trial_centre.Centre_id
where Trial_type.trial_type_name='" & Me.DropDownList1.Text & "'",
con)
        Dim ds As New DataSet
        da.Fill(ds, "tab")
        Dim oRpt As New
CrystalDecisions.CrystalReports.Engine.ReportDocument
        ''oRpt.Load(Server.MapPath("CrReport.rpt"))
        oRpt.Load(Server.MapPath("CrReport.rpt"))
        oRpt.SetDataSource(ds)

        'str1 = "Resource Management Trial"
        'str2 = "ZXCDVG"
        'str3 = "Rural4"
        'str4 = "2010"
        ''str1 = Me.DropDownList1.Text
        '---------------------------

        oRpt.SetParameterValue("aa", str1)
        oRpt.SetParameterValue("bb", str2)
        oRpt.SetParameterValue("cc", str3)
```

```
        oRpt.SetParameterValue("dd", str4)
        oRpt.DataDefinition.FormulaFields("ChkVar").Text =
GetAlltheTC()
        oRpt.DataDefinition.FormulaFields("AllTcert").Text =
GetAlltheTCNo()
        oRpt.DataDefinition.FormulaFields("Cent").Text = GetCentre()


        '--------------------------
        CrystalReportViewer1.Visible = True
        CrystalReportViewer1.ReportSource = oRpt
        'CrystalReportViewer1.HasExportButton = True
        'CrystalReportViewer1.HasPrintButton = True
        CrystalReportViewer1.DataBind()
    End Sub
    Public Function GetAlltheTCNo() As String
        If (IsPostBack = True) Then

            Try
                Dim con As New SqlConnection
                con.ConnectionString =
"Server=.;database=ISVE;trusted_connection=true"
                Dim StrQry As String
                Dim StrResult As String
                Dim Dr As SqlDataReader
                Dim Cmd As New SqlCommand ';, TC.CertificateNo"
                con.Open()
                ''StrQry = "SELECT Variety.Variety_name AS
Variety_name FROM Var_trial_centre_variety INNER JOIN Variety ON
Var_trial_centre_variety.Variety_id = Variety.Variety_id where
Var_trial_centre_variety.varital_trial_id='5'"
                ''Cmd.CommandText = StrQry

                ''Dr = Cmd.ExecuteReader
                Dim cmd1 As New SqlCommand("SELECT
Variety.Variety_name AS Variety_name FROM Var_trial_centre_variety
INNER JOIN Variety ON Var_trial_centre_variety.Variety_id =
Variety.Variety_id where
Var_trial_centre_variety.varital_trial_id='" & Me.HiddenField1.Value
& "'", con)
                ''Dim dr As SqlDataReader
                Dr = cmd1.ExecuteReader
                If Dr.HasRows = False Then Dr.Close() : Exit
Function
                Do While Dr.Read
                    If GetAlltheTCNo = "" Then
                        GetAlltheTCNo = Dr("Variety_name")
                    Else
                        GetAlltheTCNo = GetAlltheTCNo & "," &
Dr("Variety_name")
                    End If
                Loop
                Dr.Close()
                If Trim(GetAlltheTCNo) = "" Then Exit Function
                GetAlltheTCNo = "'" & " " & GetAlltheTCNo & "'"
                con.Close()

            Catch ex As Exception

            End Try
```

```
        End If

    End Function

    Protected Sub Page_Load(ByVal sender As Object, ByVal e As
System.EventArgs) Handles Me.Load
        If IsPostBack = False Then

            FillTrialType()
            FillTrialSeries()
            '' FillTrialSeries()
            FillZone()
            FillYear()

            ''End If
        Else
            ''Me.Page.IsPostBack = False
            ''Me.CrystalReportViewer1.DisplayPage = True
            ''Me.CrystalReportViewer1.Enabled = True
            ''Button2.Attributes.Add("onclick", "return
CallPrint(strid)")



        end if


    End Sub
    Public Sub FillTrialType()
        Try
            Dim con As New SqlConnection
            con.ConnectionString =
"Server=.;database=ISVE;trusted_connection=true"
           Dim Cmd As New SqlCommand ';, TC.CertificateNo"
            con.Open()
           Dim cmd1 As New SqlCommand("SELECT  distinct
Trial_type.Trial_type_name FROM Varietal_trial INNER JOIN
Trial_series ON Varietal_trial.Trial_series_id =
Trial_series.Trial_series_id INNER JOIN Trial_type ON
Trial_series.Trial_type_id = Trial_type.Trial_type_id ", con)
            Dim dr As SqlDataReader
            Dr = cmd1.ExecuteReader
            If Dr.HasRows = False Then Dr.Close() : Exit Sub
            Do While Dr.Read
                Me.DropDownList1.Items.Add(dr(0).ToString)
            Loop
            Dr.Close()
           Catch ex As Exception
        End Try
    End Sub
    Public Sub FillTrialSeries()
        Try
            Me.DropDownList2.Items.Clear()
            Dim con As New SqlConnection
            con.ConnectionString =
"Server=.;database=ISVE;trusted_connection=true"
            Dim Cmd As New SqlCommand ';, TC.CertificateNo"
            con.Open()
            ''Dim cmd1 As New SqlCommand("SELECT  distinct
Trial_series.Trial_series_name FROM Varietal_trial INNER JOIN
```

```
Trial_series ON Varietal_trial.Trial_series_id =
Trial_series.Trial_series_id", con)
            Dim cmd1 As New SqlCommand("SELECT distinct
Trial_series.Trial_series_name FROM trial_series INNER JOIN
Varietal_trial ON Trial_series.Trial_series_id =
Varietal_trial.Trial_series_id INNER JOIN Trial_type ON
Trial_series.Trial_type_id = Trial_type.Trial_type_id where
Trial_type.trial_type_name='" & Me.DropDownList1.Text & "'", con)
            Dim dr As SqlDataReader
            dr = cmd1.ExecuteReader
            If dr.HasRows = False Then dr.Close() : Exit Sub
            Do While dr.Read
                Me.DropDownList2.Items.Add(dr(0).ToString)
            Loop
            dr.Close()
        Catch ex As Exception
        End Try
    End Sub
    Public Sub FillZone()
        Me.DropDownList3.Items.Clear()
        Try
            Dim con As New SqlConnection
            con.ConnectionString =
"Server=.;database=ISVE;trusted_connection=true"
            Dim Cmd As New SqlCommand ';, TC.CertificateNo"
            con.Open()
            ''Dim cmd1 As New SqlCommand("SELECT distinct zone_name
from zone", con)
            Dim cmd1 As New SqlCommand("SELECT DISTINCT
[Zone].Zone_name FROM Varietal_trial INNER JOIN Trial_series ON
Varietal_trial.Trial_series_id = Trial_series.Trial_series_id INNER
JOIN Trial_type ON Trial_series.Trial_type_id =
Trial_type.Trial_type_id INNER JOIN [Zone] ON Varietal_trial.zone_id
= [Zone].Zone_id WHERE Trial_type.TRIAL_TYPE_NAME='" &
Me.DropDownList1.Text & "' AND Trial_series.TRIAL_SERIES_NAME='" &
Me.DropDownList2.Text & "'", con)
            Dim dr As SqlDataReader
            dr = cmd1.ExecuteReader
            If dr.HasRows = False Then dr.Close() : Exit Sub
            Do While dr.Read
                Me.DropDownList3.Items.Add(dr(0).ToString)
            Loop
            dr.Close()
        Catch ex As Exception
        End Try
    End Sub
    Public Sub FillYear()
        Me.DropDownList4.Items.Clear()
        Try

            Dim con As New SqlConnection
            con.ConnectionString =
"Server=.;database=ISVE;trusted_connection=true"
            Dim Cmd As New SqlCommand ';, TC.CertificateNo"
            con.Open()
            Dim cmd1 As New SqlCommand("SELECT  DISTINCT
Varietal_trial.[Year] FROM Trial_type INNER JOIN Trial_series ON
Trial_type.Trial_type_id = Trial_series.Trial_type_id INNER JOIN
Varietal_trial ON Trial_series.Trial_series_id =
Varietal_trial.Trial_series_id INNER JOIN [Zone] ON
Varietal_trial.zone_id = [Zone].Zone_id WHERE
```

```
Trial_type.TRIAL_TYPE_NAME='" & Me.DropDownList1.Text & "' AND
Trial_series.TRIAL_SERIES_NAME='" & Me.DropDownList2.Text & "' AND
ZONE.ZONE_NAME='" & Me.DropDownList3.Text & "'", con)
            Dim dr As SqlDataReader
            dr = cmd1.ExecuteReader
            If dr.HasRows = False Then dr.Close() : Exit Sub
            Do While dr.Read
                Me.DropDownList4.Items.Add(dr(0).ToString)
            Loop
            dr.Close()
        Catch ex As Exception
        End Try
    End Sub


    Private Sub SearchVID()
        If (IsPostBack = True) Then

            Try
                Dim con As New SqlConnection
                con.ConnectionString =
"Server=.;database=ISVE;trusted_connection=true"

                Dim Dr As SqlDataReader
                Dim Cmd As New SqlCommand ';, TC.CertificateNo"
                con.Open()

                Dim cmd1 As New SqlCommand("SELECT distinct
Varietal_trial.Varietal_trial_id FROM Trial_series INNER JOIN
Trial_type ON Trial_series.Trial_type_id = Trial_type.Trial_type_id
INNER JOIN Varietal_trial ON Trial_series.Trial_series_id =
Varietal_trial.Trial_series_id INNER JOIN Var_tri_CTL ON
Varietal_trial.Varietal_trial_id = Var_tri_CTL.Varietal_trial_id
INNER JOIN Control_variety ON Var_tri_CTL.Control_variety_id =
Control_variety.control_variety_id INNER JOIN [Zone] ON
Varietal_trial.zone_id = [Zone].Zone_id where
Trial_type.Trial_type_name='" & Me.DropDownList1.Text & "'  and
Trial_series.Trial_series_name='" & Me.DropDownList2.Text & "' and
Varietal_trial.year='" & Me.DropDownList4.Text & "'  and
[Zone].Zone_name='" & Me.DropDownList3.Text & "'", con)

                Dr = cmd1.ExecuteReader
                If Dr.HasRows = False Then Dr.Close() : Exit Sub
                Do While Dr.Read
                    Me.HiddenField1.Value = Dr("Varietal_trial_id")
                Loop
                Dr.Close()


            Catch ex As Exception

            End Try
        End If

    End Sub
    Public Function GetAlltheTC() As String
        If (IsPostBack = True) Then

            Try
                Dim con As New SqlConnection
```

```
                con.ConnectionString =
"Server=.;database=ISVE;trusted_connection=true"
                Dim StrQry As String
                Dim StrResult As String
                Dim Dr As SqlDataReader
                Dim Cmd As New SqlCommand ';, TC.CertificateNo"
                con.Open()
                ''StrQry = "SELECT Variety.Variety_name AS
Variety_name FROM Var_trial_centre_variety INNER JOIN Variety ON
Var_trial_centre_variety.Variety_id = Variety.Variety_id where
Var_trial_centre_variety.varital_trial_id='5'"
                ''Cmd.CommandText = StrQry

                ''Dr = Cmd.ExecuteReader
                ''Dim cmd1 As New SqlCommand("SELECT
Control_variety.control_variety_Name FROM Trial_series INNER JOIN
Trial_type ON Trial_series.Trial_type_id = Trial_type.Trial_type_id
INNER JOIN Varietal_trial ON Trial_series.Trial_series_id =
Varietal_trial.Trial_series_id INNER JOIN Var_tri_CTL ON
Varietal_trial.Varietal_trial_id = Var_tri_CTL.Varietal_trial_id
INNER JOIN Control_variety ON Var_tri_CTL.Control_variety_id =
Control_variety.control_variety_id INNER JOIN [Zone] ON
Varietal_trial.zone_id = [Zone].Zone_id where
Trial_type.Trial_type_name='" & Me.DropDownList1.Text & "' and
Trial_series.Trial_series_name='" & Me.DropDownList2.Text & "' and
Varietal_trial.year='" & Me.DropDownList4.Text & "' and
[Zone].Zone_name='" & Me.DropDownList3.Text & "'", con)
                Dim cmd1 As New SqlCommand("SELECT
Control_variety.control_variety_Name FROM Trial_series INNER JOIN
Trial_type ON Trial_series.Trial_type_id = Trial_type.Trial_type_id
INNER JOIN Varietal_trial ON Trial_series.Trial_series_id =
Varietal_trial.Trial_series_id INNER JOIN Var_tri_CTL ON
Varietal_trial.Varietal_trial_id = Var_tri_CTL.Varietal_trial_id
INNER JOIN Control_variety ON Var_tri_CTL.Control_variety_id =
Control_variety.control_variety_id INNER JOIN [Zone] ON
Varietal_trial.zone_id = [Zone].Zone_id where
Varietal_trial.Varietal_trial_id='" & Me.HiddenField1.Value & "'",
con)
                ''Dim dr As SqlDataReader
                Dr = cmd1.ExecuteReader
                If Dr.HasRows = False Then Dr.Close() : Exit
Function
                Do While Dr.Read
                    If GetAlltheTC = "" Then
                        GetAlltheTC = Dr("control_variety_Name")
                    Else
                        GetAlltheTC = GetAlltheTC & "," &
Dr("control_variety_Name")
                    End If
                Loop
                Dr.Close()
                If Trim(GetAlltheTC) = "" Then Exit Function
                GetAlltheTC = "'" & " " & GetAlltheTC & "'"
                con.Close()

            Catch ex As Exception

            End Try

        End If
```

```
    End Function
    Public Function GetCentre() As String
        If (IsPostBack = True) Then

            Try
                Dim con As New SqlConnection
                con.ConnectionString =
"Server=.;database=ISVE;trusted_connection=true"
                Dim StrQry As String
                Dim StrResult As String
                Dim Dr As SqlDataReader
                Dim Cmd As New SqlCommand ';, TC.CertificateNo"
                con.Open()
                ''StrQry = "SELECT Variety.Variety_name AS
Variety_name FROM Var_trial_centre_variety INNER JOIN Variety ON
Var_trial_centre_variety.Variety_id = Variety.Variety_id where
Var_trial_centre_variety.varital_trial_id='5'"
                ''Cmd.CommandText = StrQry

                ''Dr = Cmd.ExecuteReader
                Dim cmd1 As New SqlCommand("SELECT
centre.Center_Name FROM  Var_trial_centre_variety INNER JOIN Variety
ON Var_trial_centre_variety.Variety_id = Variety.Variety_id INNER
JOIN  centre ON Variety.Centre_id = centre.Centre_id INNER JOIN
Varietal_trial ON Var_trial_centre_variety.Varital_trial_id =
Varietal_trial.Varietal_trial_id where
Varietal_trial.Varietal_trial_id='" & Me.HiddenField1.Value & "'",
con)
                ''Dim dr As SqlDataReader
                Dr = cmd1.ExecuteReader
                If Dr.HasRows = False Then Dr.Close() : Exit
Function
                Do While Dr.Read
                    If GetCentre = "" Then
                        GetCentre = Dr("Center_Name")
                    Else
                        GetCentre = GetCentre & "," &
Dr("Center_Name")
                    End If
                Loop
                Dr.Close()
                If Trim(GetCentre) = "" Then Exit Function
                GetCentre = "'" & " " & GetCentre & "'"
                con.Close()

            Catch ex As Exception

            End Try

        End If

    End Function

    Protected Sub Button2_Click(ByVal sender As Object, ByVal e As
System.EventArgs) Handles Button2.Click

        Button2.Attributes.Add("onclick", "return CallPrint(strid)")

    End Sub
```

```
    Protected Sub DropDownList1_SelectedIndexChanged(ByVal sender As
Object, ByVal e As System.EventArgs) Handles
DropDownList1.SelectedIndexChanged
        If (IsPostBack = True) Then
            FillTrialSeries()
            FillZone()
            FillYear()
        End If

    End Sub

    Protected Sub DropDownList2_SelectedIndexChanged(ByVal sender As
Object, ByVal e As System.EventArgs) Handles
DropDownList2.SelectedIndexChanged
        If (IsPostBack = True) Then
            FillZone()
            FillYear()

        End If
    End Sub

    Protected Sub DropDownList3_SelectedIndexChanged(ByVal sender As
Object, ByVal e As System.EventArgs) Handles
DropDownList3.SelectedIndexChanged
        If (IsPostBack = True) Then

            FillYear()
        End If
    End Sub
End Class
```

Code for Data Sheet Report page

```
<*********************************************************************>

using System;
using System.Data;
using System.Configuration;
using System.Collections;
using System.Web;
using System.Web.Security;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Web.UI.HtmlControls;
using System.Data.SqlClient;

public partial class Report1 : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        if (IsPostBack == false)
        {

            string constring1 = "Data source = localhost; initial
catalog=ISVE; integrated security=true;";
            string concommand1 = "SELECT  distinct
Trial_series.Trial_series_name FROM DataEntry INNER JOIN
Varietal_trial ON DataEntry.Varietal_trial_Id =
Varietal_trial.Varietal_trial_id INNER JOIN Trial_series ON
Varietal_trial.Trial_series_id = Trial_series.Trial_series_id ";
            SqlConnection con1 = new SqlConnection(constring1);
            con1.Open();
```

```
            SqlCommand cmd1 = new SqlCommand(concommand1, con1);
            SqlDataReader dr1 = cmd1.ExecuteReader();
            while (dr1.Read())
            {
                DropDownList1.Items.Add(dr1[0].ToString());
            }
            FillDesign();
            FillCharacter();
            FillCentre();
        }


        //FillTrialSeries();
    }
    public void BindData()
    {
  SqlConnection myConnection = new
SqlConnection("Server=.;database=ISVE;trusted_connection=true");

        SqlDataAdapter ad = new SqlDataAdapter("SELECT * FROM
DataEntry", myConnection);

        DataSet ds = new DataSet();

        ad.Fill(ds, "DataEntry");

        this.DataGrid1.DataSource = ds;

        this.DataGrid1.DataBind();
    }
    private void FillDesign()
    {
        this.DropDownList2.Items.Clear();
        //if (IsPostBack == false)

            string constring1 = "Data source = localhost; initial
catalog=ISVE; integrated security=true;";
            //string concommand1 = "SELECT distinct
Design.Design_name FROM Varietal_trial INNER JOIN Design ON
Varietal_trial.Design_id = Design.Design_id INNER JOIN Trial_series
ON Varietal_trial.Trial_series_id = Trial_series.Trial_series_id";
            string concommand1 = "SELECT distinct Design.Design_name
FROM DataEntry INNER JOIN Varietal_trial ON
DataEntry.Varietal_trial_Id = Varietal_trial.Varietal_trial_id INNER
JOIN Trial_series ON Varietal_trial.Trial_series_id =
Trial_series.Trial_series_id INNER JOIN Design ON
Varietal_trial.Design_id = Design.Design_id where
Trial_series.trial_series_name='" + this.DropDownList1.Text + "'";
            SqlConnection con1 = new SqlConnection(constring1);
            con1.Open();
            SqlCommand cmd1 = new SqlCommand(concommand1, con1);
            SqlDataReader dr1 = cmd1.ExecuteReader();
            while (dr1.Read())
            {
                DropDownList2.Items.Add(dr1[0].ToString());
            }

    }
    protected void Button1_Click(object sender, EventArgs e)
    {
```

```
        SqlConnection con = new
SqlConnection("Server=.;database=ISVE;trusted_connection=true");
        SqlDataAdapter da = new SqlDataAdapter("select * from
Control_variety", con);
        DataSet ds = new DataSet();
        da.Fill(ds, "Control_variety");
        CrystalDecisions.CrystalReports.Engine.ReportDocument crpt =
new CrystalDecisions.CrystalReports.Engine.ReportDocument();
        crpt.Load(Server.MapPath("CrystalReport.rpt"));
        crpt.SetDataSource(ds);

        String str1, str2, str3, str4;
        str1 = "'" + this.DropDownList1.Text + "'";
        str2 = "'" + this.DropDownList2.Text + "'";
        str3 = "'" + this.DropDownList3.Text + "'";
        str4 = "'" + this.DropDownList4.Text + "'";
        crpt.SetParameterValue("a", str1);
        crpt.SetParameterValue("b", str2);
        crpt.SetParameterValue("c", str3);
        crpt.SetParameterValue("d", str4);
        CrystalReportViewer1.Visible = true;
        CrystalReportViewer1.ReportSource = crpt;
        CrystalReportViewer1.HasExportButton = true;
        CrystalReportViewer1.DataBind();
    }
    private void FillCharacter()
    {
        this.DropDownList3.Items.Clear();
        //if (IsPostBack == false)
        //{
            string constring1 = "Data source = localhost; initial
catalog=ISVE; integrated security=true;";
            //string concommand1 = "select distinct Charachter from
DataEntry";
            string concommand1 = "SELECT distinct
DataEntry.Charachter FROM DataEntry INNER JOIN  Varietal_trial ON
DataEntry.Varietal_trial_Id = Varietal_trial.Varietal_trial_id INNER
JOIN Trial_series ON Varietal_trial.Trial_series_id =
Trial_series.Trial_series_id INNER JOIN Design ON
Varietal_trial.Design_id = Design.Design_id where
Trial_series.trial_series_name='" + this.DropDownList1.Text + "' and
Design.Design_name='" + this.DropDownList2.Text + "'";
            SqlConnection con1 = new SqlConnection(constring1);
            con1.Open();
            SqlCommand cmd1 = new SqlCommand(concommand1, con1);
            SqlDataReader dr1 = cmd1.ExecuteReader();
            while (dr1.Read())
            {
                DropDownList3.Items.Add(dr1[0].ToString());
            }

    }

    protected void DropDownList1_SelectedIndexChanged(object sender,
EventArgs e)
    {
        if (IsPostBack == true)
        {

            FillDesign();
            FillCharacter();
```

```
            FillCentre();
        }
    }
    private void FillCentre()
    {
        this.DropDownList4.Items.Clear();

            string constring1 = "Data source = localhost; initial
catalog=ISVE; integrated security=true;";
            //string concommand1 = "select distinct Centre from
DataEntry";
            string concommand1 = "SELECT distinct DataEntry.Centre
FROM DataEntry INNER JOIN Varietal_trial ON
DataEntry.Varietal_trial_Id = Varietal_trial.Varietal_trial_id INNER
JOIN Trial_series ON Varietal_trial.Trial_series_id =
Trial_series.Trial_series_id INNER JOIN Design ON
Varietal_trial.Design_id = Design.Design_id where
Trial_series.trial_series_name='" + this.DropDownList1.Text + "' and
Design.Design_name='" + this.DropDownList2.Text + "' and
DataEntry.charachter='" + this.DropDownList3.Text + "'";
            SqlConnection con1 = new SqlConnection(constring1);
            con1.Open();
            SqlCommand cmd1 = new SqlCommand(concommand1, con1);
            SqlDataReader dr1 = cmd1.ExecuteReader();
            while (dr1.Read())
            {
                DropDownList4.Items.Add(dr1[0].ToString());
            }

    }
    protected void DropDownList3_SelectedIndexChanged(object sender,
EventArgs e)
    {
        if (IsPostBack == true)
        {
            FillCentre();
        }
    }
    protected void DropDownList2_SelectedIndexChanged(object sender,
EventArgs e)
    {
        if (IsPostBack == true)
        {
            FillCharacter();
            FillCentre();
        }
}

    protected void DropDownList4_SelectedIndexChanged(object sender,
EventArgs e)
    {

    }
}
```