

# **A PROJECT REPORT ON GENEBANK INFORMATION SYSTEMS**

**T.V.SESHAGIRI RAO**



**WANAPARTHY COLLEGE OF COMPUTERS & MANAGEMENT**

**(Affiliated to Osmania University)**

**1999**



**International Crops Research Institute for the Semi-Arid Tropics**

eru 502 324  
1ra Pradesh  
India

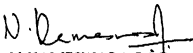
Tel +91 40 3296161 (19 lines)  
Fax +91 40 241239  
+91 40 3296182  
Email ICRISAT@CGIAR.ORG

CERTIFICATE

This is to certify that the dissertation entitled "Genebank Information Systems" is a bonafied record of the work done by SESHAGIRI RAO, T.V. at ICRISAT, Patancheru under our supervision in partial fulfilment for the award of "Master of Computer Applications", Osmania University, Hyderabad.

The results embedded in this dissertation is not submitted to any other University or Institute for the award of any other degree or diploma.

  
B. DIWAKAR  
Senior Manager  
Training & Fellowships Program  
ICRISAT, Patancheru  
Andhra Pradesh, INDIA

  
N. KAMESWARA RAO  
Genebank Curator  
GREP  
ICRISAT, Patancheru  
Andhra Pradesh, INDIA

Visit our worldwide web site at <http://www.icrisat.org>

IT is part of the global agricultural research network called the Consultative Group on International Agricultural Research (CGIAR)



# **WANAPARTHY COLLEGE OF COMPUTERS & MANAGEMENT**

(P.G. College, MCA, Affiliated to Omania University)

H.NO. 37-60, VALLABH NAGAR, (N.C.C.T.) WANAPARTHY - 509 103.  
MAHABUBNAGAR (DIST.) A.P.

PHONES · OFFI : 22312 STD (08545), COLLEGE . 22303

DEPARTMENT OF COMPUTER SCIENCE

## **CERTIFICATE**

This is to certify that the project work entitled GENE BANK INFORMATION SYSTEMS is a bonafied record of work done by T.V.SESHAGIRI RAO, ht. 2997026 as a partial fulfillment for the award of the postgraduate Degree Master Of Computer Applications during the VI Semester of the course

**Internal Guide**

**External Examiner**

## **Acknowledgements**

I express deep sense of gratitude and sincere thanks to Dr. N. Kameswara Rao, Genebank curator, GREP for his supervision, co-operation and encouragement during the course of the endeavor.

I am equally grateful to Dr.V.Mahalakshmi, senior scientist, Genetic Resources and Enhancement Program for her guidance and Dr.B.Diwakar, senior manager, Training and Fellowships Program for his help.

I am sincerely thankful to Mr.V.Gopal Reddy, scientific officer, Mr.K.Narashimha Reddy, scientific officer, Mr.D.V.S.S.R.Sastry, scientific officer, Mr.B.Ashok Kumar, administrative associate and Mr.M.Thimma Reddy, research technician for their technical contribution and continuous help during my course work.

I am thankful to Mr.K.Upendra, Mr.S.A.Arshad, and Mr.M.Raja Ganesh for their full cooperation towards completion of this work.

I wish to express my gratitude to Mr.S.VijayamohanRao, Correspondent, Computer faculty members Mr.Ramana Murthy, Mr.Rambabu and other staff members.

I wish to utilize this opportunity to acknowledge with sincere appreciation for the continuous encouragement received and for the services extended by different people and especially thankful to my parents for their encouragement.

**T.V.Seshagirao.**



# Contents

# Contents

- 1. About The Organization**
- 2. About The Software**
  - 2.1 Graphical User Interface**
  - 2.2 An Overview of VisualBasic 6**
    - 2.2.1 why Visual Basic only**
    - 2.2.2 Visual Basic controls**
    - 2.2.3 Introduction to Data Access with Visual Basic**
    - 2.2.4 Database Management Using Visual Basic**
    - 2.2.5 Introduction to Crystal Reports**
    - 2.2.6 Visual Basic Active X Controls**
    - 2.2.7 Feasibility Study**
    - 2.2.8 Advantages Of RDBMS**
    - 2.2.9 Disadvantages Of RDBMS**
    - 2.2.10 System Implementation**
    - 2.2.11 System Testing and Maintenance**
  - 2.3 Why SQL Server**
- 3. An Overview Of ICRISAT Genebank**
- 4. System Design And Development**
- 5. System Testing**
- 6. Software Specification**
- 7. Hardware Specification**
- 8. Data Dictionary**
- 9. Data Flow Diagrams**
- 10. Screens**

**11. Reports**

**12. Bibliography**

# About the organization

## **About ICRISAT**

The Semi-Arid Tropics (SAT) encompasses parts of 48 developing countries including most of India, parts of Southeast Asia, and a swathe across sub-Saharan Africa, much of southern and eastern Africa, and parts of Latin America. Many of these countries are among the poorest in the world. Approximately one sixth of world's population lives in the SAT, which is typified by unpredictable weather, limited and erratic rainfall and nutrient-poor soil.

ICRISAT was established in 1972 with a mandate to improve agricultural productivity in SAT. It is one of the 16-non profit, research and training centers funded through the Consultative Group on International Agricultural Research (CGIAR). The CGIAR is an informal association of approximately 50 public and private sector donors. It is co-sponsored by the Food and Agriculture Organization of the United Nations (FAO), the World Bank and the United Nations Development Program (UNDP).

The ICRISAT is located at Patancheru, Andhra Pradesh, India. It is representative of the Semi-Arid Tropical (SAT) environment, and consists of two major soil orders found in the SAT. In 1972, the Government of India (GOI) made this land available to ICRISAT to establish its international headquarters. The Government of India in 1973 rehabilitated the two villages that were within the boundaries of ICRISAT.

### **ICRISAT's Mandate:**

- Serve as a world center for the improvement of grain yield and quality of sorghum, pearl millet, chickpea, pigeonpea and groundnut and act as a world repository for the genetic resources of these crops.
- Develop improved farming systems that will help to increase and stabilize agricultural production through more effective use of natural and human resources in the seasonally dry Semi Arid Tropics.
- Identify constraints to agricultural development in the Semi -Arid Tropics and evaluate means of alleviating them through technological and institutional changes.

- Assist in the development and transfer of technology to the farmer through co-operation with national and regional research programs and by sponsoring workshops and conferences, operating training programs and assisting extension activities.

### **ICRISAT's Crops:**

ICRISAT's mandate crops are sorghum, pearl millet chickpea, pigeonpea and groundnut.

- Sorghum: Cultivated throughout the semi-arid tropics and is a major source of food and fodder.
- Pearl millet: Probably, the world's hardiest crop, it is the food staple in the driest parts of the semi-arid tropics.
- Chickpea: A traditional source of protein for people in Asia and North Africa. Its importance is increasing in Europe, the Americas and Oceania.
- Pigeonpea: A staple food for South Asians, it is fast becoming an important legume in Africa.
- Groundnut: Also known as peanut, grows in a range of climatic conditions. It is consumed as food and edible oil and used as fodder.

### **Training facilities at ICRISAT:**

ICRISAT is committed to help resource-poor farmers of Semi-Arid Tropics (SAT). National Agricultural Research Systems (NARS) in the region face the constant challenge of developing skilled agricultural staff capable of transferring new technologies to farmers to help them raise their crops in harsh climatic conditions. The Training and Fellowship Program (TAFP) helps NARS to meet this challenge by providing training opportunities of various types that better equip national program staff to help the farmers.

Since 1974 ICRISAT has trained several scientists and technicians from over 91 countries. TAFP provides on the job training research opportunities, short-term technology-oriented courses and technical study programs in subject's relevant to ICRISAT's mandate. To meet the special needs of individuals working in NARS or co-

operating institutions, practical training in specific areas is provided for junior scientists, technicians, research station managers and other staff.

Apprenticeships of one to six months are offered to degree or diploma candidate's in completing a B.Sc., Ms. and MCA or from vocational agricultural facilities. Practical experience is provided to apprenticeships that work through the institute, in such areas as research, administration, computer services, information and library Services, Farm services and engineering.

## ABOUT THE SOFTWARE

### Graphical User Interface (GUI)

Introduction:

Graphical User Interfaces (GUIs) offer a standard and feel to application thus Reducing development and learning time. A GUI is an application environment that can Work with graphical objects. For example Microsoft Windows has the following

#### ***Components.***

- *Menus (including placement and names of positions and style such as pull down or pop).*
- *Icons (for identifying applications and resources).*
- *Tiled windows (for views of multiple programs or for multiple views of a single Program or data block).*
- *Dialog boxes for selecting files , options and settings.when an option is selected ,the One previously turned off.*
- *Checklists from which the user can make multiple selections as specifying print or File attributes.*
- *Support for printing device, typically a mouse (especially to select and drag screen Elements).*
- *Scroll bars along the edges of windows to show the relative position of the contents (Such as the end or beginning if the text) or to move to a different position (such as Another part of spreadsheet).*

A GUI enforces consistency by restricting developers- they must support features of the program to run under the GUI. In addition, suggestions from the GUI's creators (such as the arrangement of menu options) often become defects standards for applications.



We have described how consistency simplifies the learning of a new application. One benefit of a GUI is that user can learn a second application faster because He or she is familiar with the environment. Note that the benefit comes succeeding Applications.

Consistency and familiarity help produce shorter learning curves. User Generally prefer the interface style they know weather it be Macintosh, Microsoft Windows or Lotus 1-2-3. The consistency offered by a GUI trades on the User's familiarity With the environment Drawing and CAD programs are the best suited to GUIs since, by there Nature, they manipulate objects, lines and curves and fill closed areas with color. For Database programs such manipulation is not as useful. However they can use GUIs Effectively to

- Specify data fields when setting up reports
- Select sort keys
- Transfer data to or from other applications (such as spreadsheet)

The last point is particularly important. Database applications often must transfer data to or from spreadsheets, word processors, desktop publishing programs business or presentation graphics programs, and static programs or project management software. GUIs generally have data exchange features such as Microsoft Windows's Dynamic Data Exchange (DDE) to handle such transfer.

## **GUI CONCEPTS:**

Graphical user interfaces are designed to present information users in graphical Form. So that they can work with friendly and easy environment.

In general GUIs present information in rectangular areas on screen called windows. GUI displays multiple applications in different windows at the same time. These way users can see what is happening in one application while working with another. Users are allowed to perform several manipulations on windows, such as changing their size and position. Windows can contain objects that can be selected by clicking a mouse pointer on small object pictures called icons. An entire window can be minimized to become an

icon and user can restore an icon to its normal size. Windows can also contain other graphical entities (such as scroll bars, sliders and buttons) that allow users to control the contents of the window and provide additional input to the application.

Instead of requiring a user to know application command, a GUI uses an intuitive, easy-to-use shell with a pointing device as well as application windows, menus, dialog boxes, buttons and so on. Popular GUIs include Macintosh on Apple Computers, Microsoft Windows on IBM PCs and compatibles, motif on UNIX machines. When compared to traditional programming, the most significant difference in the presentation logic that controls GUI is the notion that the user must always be in the control of the logic. Thus traditional structured programming with its house keeping section, processing section and an output section to be modified. GUI programming should be able to accept and process asynchronous event initiated by the system at any time. This leads to the concept of event driven programming.

### Event driven programming:

To understand what event-driven programming is, let us first take a look at the way a traditional linear program works.

A linear program has a direct linear flow of control through the program there is some input, some processing and some output. Even the most complex linear program, such as state machines still tends to be fairly straightforward. There are certain numbers of states one can enter and a limited number of operations one can perform in each state.

In event-driven program, input comes from any device at any time. The program has no way of knowing in advance whether the user is going to type character or click a button. This is possible for two reasons. First, every application is totally dependent on system (window) to provide processing time, monitors space and other resources.

Ultimately the system controls every system application. Second the nature of the window encourage system (windows) application to be event-driven. Event-driven

program reduces the modest to minimum, so the user doesn't need to navigate a deep menu and can perform any action at any time. The user, not the application can control the screen. The application simply performs some set up and then goes into a loop from which applications, functions may be invoked in any orders as events arrive.

### **Event Types:**

The set of supported user generated input and system generated events differs from one GUI implementation to another, the common type of events are:

Mouse events occur when a user has moved the mouse pointer into or out of the entity, clicked on a mouse button within or without entity or release the mouse button. Keyboard events occur when a user has pressed or released a keyboard key. Menu events occur when a user selects command from menu. Windows updates events occur when a portion of the application window has been damaged and has to be redrawn. Resizing events generated by GUI to allow user to change the current active window. Initialize terminate events occur when a GUI entity has been created or destroyed so that the application can perform necessary set-up or clean-up logic.

### **Software Environment:**

This package has been developed so as to provide a user-friendly environment for its user. The GUI concepts were employed through out the package using Visual Basic. On Windows 95 platform.

## **AN OVERVIEW OF VISUAL BASIC**

Microsoft Visual Basic - the quickest and easiest way to create applications for the Microsoft Windows Operating system. The Visual Basic programming system allows us to create attractive and useful applications that fully exploit the Graphical User Interface (GUI). Graphical User Interface or GUI's have revolutionized the microcomputer industry. Windows applications generally have a consistent user interface. This means that user can spend more time mastering the application and less time worrying about which keystrokes do what within menus and display boxes. While programmers have long had mixed feelings about GUI's, beginning users seem to like them and so Windows programming is expected to be based on the GUI model? Therefore if we need to develop programs for any version of windows, we will want a tool to develop GUI based application efficiently. For a long time there were few such tools for developing Windows applications. Before Visual Basic was introduced in 1991, developing window applications was much harder than developing Dos applications. Programmers had too much worry about, such as what the mouse was doing, where the user inside menu.

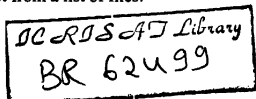
Visual Basic makes us more productive by providing appropriate tools for the different aspects of GUI development. We create the GUI for our application drawing objects in a graphical way. We set properties of these objects to refine their appearance and behavior. Using Visual Basic we can create powerful, full-featured applications that exploit the features of Microsoft Windows, including Multiple Document interface (MDI), Object Linking and Embedding (OLE), Dynamic Data Exchange (DDE), Graphics etc., and Visual Basic can be extended by adding custom controls and by calling

Procedures in a dynamic link libraries (DLL). We can finish the applications a true (.EXE) file those use a run time DLL we can freely distribute. In particular Visual Basic lets us

To add menus, text boxes, command buttons, option buttons, check boxes, list boxes, scroll bars and file and directory boxes to blank windows. We can use grids to handle tabular data and we can access databases.

## VISUAL BASIC CONTROLS:

- Pointer** : Provides a way to move and resize forms and controls.
- Picture Box** : Displays bitmaps, icons, window metafiles, it provides  
An in what to display text or acts as Visual container of  
the other controls.
- Label** : Displays text a user cannot interact with or modify.
- Text Box** : Provides an area to input or display text.
- Frame** : Provides an area to input or display text.
- Command Button**: Carries out a command or action when a user chooses it.
- Check Box** : Displays true/false or yes/no option. Any number check  
boxes on a form can be checked at one time.
- Option Button** : As a part of an option group with other option buttons,  
Display multiple choices from which a user can select  
Only one.
- Combo Box** : Combines a text box with a list box. Allows a user to  
type in selection or select an item from drop-down list.
- List box** : Displays a list of items that a user can choose from list.
- Horz Scrl Bar** : Allows a user to select a value with in a range of values.
- Vert Scrl Bar** : Allows a user to select a value with in a range of values.
- Timer** : Executes timer events at specified timer intervals.
- Drive List Box** : Displays and allows a user to select valid disk drives.
- Dir List Box** : Displays and allows a user to select from a list of files.
- File List Box** : Displays and allows a user to select from a list of files.



- Shape** : Adds a rectangular, square, ellipse or circle to a form.
- Line** : Adds a straight-line segment to a form.
- Image** : Display bitmaps, icons or windows metafiles acts like a Command button when clicked.
- Data Control** : Enables you to connect to an existing database and display Information from it on your forms.

## **Introduction to Data Access With Visual Basic:**

Vitality all applications written are used to gather information and store it for later retrieval. More and more this data is managed using the formal structure of a database. The Microsoft Visual Basic programming system for the windows operating system now enables you to create databases and to build applications that serve as front-ends to many of the popular database formats. By adding built in data. Visual Basic makes database management principle simple, fast and trouble free. Visual Basic implements data access by incorporating the same rdo engine that power data from SQL Server. Combined with Visual Basic this technology gives you seem less access to many standard database formats. Which includes Microsoft Access, Btrieve, Dbase, and Microsoft Foxpro, Oracle, Paradox and Microsoft SQL Server. If we have an existing database created in one of those supported formats specified in the above we can use Visual Basic to manipulate databases with in a minimum of programming. The data object model simplifies the code to write and insulates us from the underling structure and techniques of retrieving and updating data. We can use the same data access objects and the same properties.

The Visual Basic database architecture is extensible, so we can use additional database drivers as developed. And with the flexibility of the database object model, we will need to make few or no changes in our code to implement them. Visual Basic no longer depends upon Open Database Connectivity (ODBC) as its sole means of database of connectivity. ODBC access is still a viable means of connecting to external databases from Visual Basic. ODBC is used to connect to Microsoft SQL Server and Oracle database management systems.

### **The Database Access:**

The database control can significantly reduce the amount of code to write for an application.



### **Data Access Resources:**

Visual Basic enables us to manipulate the data and underlying data structures of many types of database applications. Visual Basic also provides access to several other popular database formats. There are three categories of databases that Visual Basic recognizes:

#### 1. Microsoft Access-format Databases:

These database files are manipulated directly Visual Basic or Microsoft Access. This is the Visual Basic native format-it provides the most flexibility and speed.

#### 2. Internal Databases:

This category includes database formats such as Btrieve, Dbase III, dBase IV, and Microsoft paradox. We can create or manipulate all of these database formats in Visual Basic.

#### 3. External databases:

This category includes client/server databases such as the Microsoft SQL Server and Oracle database management system. In the case of the ODBC database we can pass SQL commands directly to the external server of processing.

### **Database Management Using Visual Basic:**

This mainly focuses on the Data Definition Language (DDL). We can use to define database structures. It explains how can use of functions, statements, methods, properties and data access objects to define our database tables, fields and indexes. It includes mainly the following:

- A. Opening a Database
- B. Closing a Database
- C. Creating a Database
- D. Modifying the structure of a Database
- E. Managing a Database

## **A. Opening a Database :**

Suppose if we have already a database that we want to use with Visual Basic, we can use two different methods to open a database in Visual Basic. They are

1. Use the Data Control
2. Use the Open Database function

Using the data control can significantly reduce the amount of code we need to open and manipulate a database. The data control imposes few limitations on the functionality, flexibility or robustness of our applications. That is why we can use the data control in most situations. If you choose not to use the data control to open a database, we will not be able to use bound controls to display or edit our data, as each bound control must be assigned to a data control. There are also restrictions on using the data control with the open database function.

We may not be able to open a database if:

The database is already opened by another user in exclusive mode.

A specific database file cannot be found.

We do not have permission based on the database's or the network's security settings.

## **B. Closing a Database:**

Once if we are using a database we must close it. Closing Database releases the resources that it holds, releases any locks or other shared resources that the database was holding, and rolls back any uncommitted transactions. If we used the open database function to open our closing the database with the close method. And note that closing database variable is especially important if we use global or static database variables.

We do not to explicitly close a database or records opened with the data control as these databases are closed automatically when you unload the form upon which the data control resides.

### **C. Creating a Database:**

If we do not already have a database, we can use Visual Basic to create one. These are four different approaches we can take to create a database.

- Use the Data Manager Application. With the Data Manager we can create Visual Basic/MS-Access native format database.
- Use the create database function, we might use create database when we want to provide a greater depth of programmatic control over a database or when we want the application itself able to create a new database at run time. The create database function supports only the Visual Basic/MS-Access native format.
- Use MS-Access or an external database or ODBC application to create a new external database. So we can create a new external database by creating a directory on our disk.
- Use the open database function to create a new external database.

### **D. Modifying the structures of a database :**

Once we have created a database, or have a database that needs to be modified, we can use the Visual Basic to change the structure. Before we attempt to modify a database structure, we must begin by closing any open record set variables. We can add new table def objects to a database, add new field and index objects to existing tables. We can also delete a tabledef from a database or delete an index from a tabledef.

### **E. Managing a database:**

When we first start Visual Basic at design time, first load one of our applications at run time, we need to make sure that either program has access to information in an initialization (. INI) file. The initialization file sets parameters that control the environment for the application.

VB provides two statements names set data access option and set default workspace hat we can use to control the initialization of the database engine.

## **INTRODUCTION TO CRYSTAL REPORTS:**

Crystal reports is a powerful yet easy to use program for creating custom reports, lists and labels using data from your existing databases. Crystal reports was designed to work with all kinds of data: numbers, currency, text, dates and Boolean fields. It has a wide range of built in tools that you can use to manipulate that data to fit your needs. Using these tools, you can:

- Make calculations and comparisons of data values.

- Calculates sub totals and grand totals of field values.

- Calculate group averages count the records in a group, and test for minimum and maximum values.

- Test for the presence of specific values.

- Present data only if certain conditions are met.

- Evaluate logical relationships between values.

- Convert data from one type to another.

- Merge text with another text.

- Perform numerous other useful data related activities.

### **Kinds of reports**

#### **Lists and reports:**

You can create simple columnar lists from a single database or you can create a report that includes data from different sources, that subjects the data to extensive calculations, that merges text with data and that highlights key data with display fonts and other character attributes.

### Cross-Tab:

You can create cross-tab reports that enable you to make comparison and identify trends in a hurry.

### Graphs/Charts:

you can create graphs and charts in a variety of formats to help you present elements of your report in a more visual manner.

### Label-Type items:

You can create mailing labels with your data and you can also create nametags, rotary file cards, disk labels and other similar label type items that are available for dot matrix or laser printers.

### Data analysis:

You can analyze mountains of data in multiple databases. You can assign priority numbers to alternatives or flag in the items of greatest interest.

### Form letters:

You can create customized form letter that simply draw data from a database for address and salutation, or you can create sophisticated letters that include different blocks of text depending on relationship found in the data.

### Invoices:

You can create custom invoices to be printed as a batch on specific dates or construct Point of sales system that generates an invoice and demand, once the appropriate data is entered.

### SECTIONS:

#### A title section:

This section is generally used for the report title and other information you want to

appear at the top of the page.

A Page Header section:

This section is generally used for field headings and other information that you want to appear at the top of each page.

A Detail Section:

This section is the body of the report. The bulk of your report data will generally appear in this section.

A Page Footer Section:

This section usually contains the page number and any other information that you want to appear on the bottom of the page.

A Summary Section:

This section allows you to include a summary at the bottom of your report. You can choose to print the summary on only the first page or on all pages.

FIELDS:

Page NumberField:

Use Insert |Special Field|Page Number Field to insert a field that prints the current page number.

Record Number Field:

Use Insert |Special Field|Record Number Field to have crystal reports number each record printed in the detail section of your report.

Print Date Field:

Use Insert |Special Field|Print date Field to insert a field that prints whatever is the current data when the report prints.

## WHAT IS ACTIVE X?

Active X is a set of technologies that integrate software components in a networked environment, regardless of the language in which they were created. This integration of components enables content and software developers to easily create interactive applications and web sites. As a leading commercial object model, active X has been widely adopted by corporate MIS and ISV communities and is used by millions of applications and content developers today.

Active X technologies make it easy to create, integrate and reuse software components, or “controls”, over the Internet or Intranets. Active X control is a software component that can be integrated into web pages, Microsoft office, Microsoft Access and Visual Basic (ie.any host that supports Active X Controls).

With Active X, developers can create components in any programming language integrate them with any scripting language and run those components from any type of application, including Web Browsers and many of the world’s most popular business applications. Assembling Web sites from a wide variety of existing software component speeds time to market allows web procedures to rebuild more engaging and effective sites and results in a more intriguing and productive experience for web surfaces.

### Active X Creation tools:

Before the Visual Basic 5.0 control creation edition, Active X components could only be created with the c \ c++ languages. The Visual Basic 5.0 control creation edition represents a powerful new approach to creating Active X Controls. Using the same Visual metaphor for building controls as it does applications, the Visual Basic control creation edition allows controls to be creating from scratch, by modifying existing controls or by assembling multiple controls.

The ability to create new controls from existing controls gives Visual Basic programmers an astonishing head start when developing new, specialized components. Rather than

starting from scratch, programs can use Visual Basic control creation edition to customize any of the 2000 or more commercially available Active X controls.

Active X controls can be hosted in a wide variety of developments tools. In a Microsoft Office 97, Active X controls can be placed on Microsoft Office Documents. Using the Active X control pad, Internet studio or front page, Active X controls can also be places on the web pages. When Active X controls are placed on the web pages, they behave like Java applets. Like Java applets, if the user surfing the web page does not have the Active X control already on their machine it is automatically down loaded. Likewise, if the user has an out dated control, the newer control will get automatically downloaded.

### FEASIBILITY STUDY:

Feasibility study encompasses the following three things.

1. Technical feasibility.
2. Operational feasibility.
3. Economic feasibility.

#### Technical feasibility:

Technical feasibility study determines weather the organization has the technology and skills necessary to carryout the project and if not how should these is obtained. Coming to our proposed system, the existing technology suffices the need of the existing system. That required hardware and software for the developments of the system are available. So we can conclude that the system is technically feasible,

#### Operational feasibility:

Operational feasibility study determines if the proposed system satisfied user objectives and can be fitted into current system operation. Our proposed system will certainly satisfy the user objectives and it will also enhance their capability. The proposed system can be best fitted into current operation. Also there is no need to replace any of the existing staff. So the system is operationally feasible.



### Economically feasibility:

Economic feasibility study determines whether the project's goal can be within the resource limits allocated to it. It must determine whether it is worthwhile to proceed with the project at all or whether the benefits obtained from the new system is not worth the costs.

After conducting cost benefit analysis, it reveals that the objectives of the proposed system can be achieved within the allocated resources. Proposed system requires no much extra manpower cost is almost nil. Also, the cash invested to implement the proposed system can be easily recovered. So the system is economically feasible.

### ADVANTAGES OF RDBMS:

1. Redundancy can be reduced :

In a non-database system, each application has its own private files. This often leads to considerable redundancy in stored data, with resultant storage in storage space.

2. Inconsistency can be avoided :

There will be some occasions on which two entries do not agree. This is called inconsistency. It can be avoided in a database by propagating updates.

3. Data can be shared :

It means that not only the existing application can share the data in the database, but also new application can be developed.

4. Standards can be enforced:

With central control of data, certain standards like industrial, national, international can be enforced.

5. Security Restrictions can be applied:

Having complete control of data, we can ensure that the only means of accessing data is through proper channel.

6. Integrity can be maintained:

The problem of integrity is the problem ensuring the data in the database is accurate. It can ensure by defining validation procedures whenever updating operations are to be carried out.

7. Conflicting requirements can be balanced:

Database can be structured to provide an overall service.

**DISADVANTAGES OF RDBMS:**

1. Major disadvantage is that it can be expensive.
2. DBMS can occupy much more main memory, that additional memory must be Purchased, thus forcing user to upgrade to a more powerful computer.
3. Large amounts of data in many different formats can be inter-related in the database. Database system and application programs must be able to process these structures.
4. Backup and recovery are more difficult in the database environment because databases are often processed by several users concurrently.
5. Centralization increases vulnerability. A failure in one component of an integrated system can stop the entire system.

IMPLEMENTATION

**SYSTEM IMPLEMENTATION:**

The implementation of application software development is concerned with the information of design specifications that have been specified at design phase, into a fully functioning database system that operates on a particular machine under the control of database management system. The primary goal of implementation is to write source code and internal conformance of the code to its specification can be easily verified and so that debugging, testing and modification are eased.

Since the design of the application software be made independent of particular databases, suitable software it to chosen and for the development of application so that transformation of design into functioning system would be efficient and easy.

A good implementation should be able to reflect good design decision. The levels of abstraction that has been employed at design phase should also be concerned at the time of implementation and also the inter-relationship between different modules should be taken care of. Availability of software for the application development is one of the essential factors that are to be considered at the time of implementation. This chapter gives the brief introduction of Visual Basic and the Data Access with Visual Basic.

#### SYSTEM TESTING AND MAINTENANCE:

The main purpose of testing is to discover any errors that may have arisen during design and implementation phase. Errors are nothing but the deviation from the behavior stipulated by the requirement analysis.

SYSTEM TESTING involves two kinds of activities namely Integration testing and Acceptance testing. In Integration testing each module, as it developed it tested solely. Once all the modules have been developed, they are integrated and tested again as a whole. Acceptance testing involves planning and execution of various types of testing in order to determine that the implemented software system satisfies the requirements stated in the requirement analysis. Both the types of testing have been made on this application, Sales Maintenance System. It has satisfied the demands that have been placed on the system by the user.

## SYSTEM MAINTENANCE:

Phase of the software life cycle is the time period in which a software product performs useful work. Maintenance activities involve making enhancements to the system. Adaptation of system to new environments Correction of errors. Enhancements to the system involve adding new functional capabilities, improving user displays, and updating the performance characteristics of the system. Adoption of the software to new environment may involve making the software to different machines. This application SMS can be run on any machine, where VISUAL BASIC is available. An exact requirement analysis is very difficult in many cases and so new user demands are often adding during the operation of the database system, which require system modifications. Corrections of errors involve modification and revalidation of software. Some errors require immediate attention, while some can be corrected on a periodic basis.

## SQL Server 7.0:

Microsoft SQL Server 7.0 Is the scalable, high performance database management system designed to meet the demanding requirements of distributed client/server computing.

Microsoft SQL Server provides:

- Integration with Microsoft Windows NT threading and scheduling services, performance Monitor, and Event Viewer. A single Windows NT logon to both the network and SQL Server simplifies management of user accounts.
- Built-in replication for reliable dissemination of information throughout an enterprise, reducing the risk of down time, and putting timely , accurate information close to people who need it.
- Parallel architecture. By executing internal database functions in parallel, the performance and scalability of the system is dramatically increased.
- Centralized management of servers throughout the enterprise with the comprehensive distributed frame work. A Windows-based management interface provides visual Drag-and-drop control over multiple servers for remote management of data replication, server administration, diagnostics and tuning.
- Better support for very large databases by taking advantages of parallel architecture. Reduces I/O for many development and maintenance tasks.
- A library of OLE Distributed Management Objects that are available In the distribute Management Framework.

One of the major challenges for client/server users is to centrally manage multiple servers across the enterprise. SQL Server 7.0 address these challenges with an enterprise system, administrative framework called the Distributed Management Framework (DMF).

The DMF is an integrated framework of objects, services, and components used to manage Microsoft SQL Server. The framework can be logically separated into three

parts: the client/front end, the SQL object library, and the Server/back end (or Windows NT Services). The client layer includes SQL Enterprise Manager as well as any custom applications written using SQL\_DMO automation objects. The objects encapsulate all of the SQL Server management functions and act as a middle layer between the client and server. The databases back end, and run as Windows NT Services.

### **SQL Enterprise Manager**

A graphical administration tool that simplifies managing a multi-server environment. This tool uses SQL-DMO automation objects.

### **SQL Distributed Management Objects (SQL-DMO)**

SQL Distributed Management Objects provide 32-bit OLE automation objects for the Microsoft Windows 95 and Microsoft Windows NT operating systems. SQL-DMO exposes interfaces for all SQL Server management functions to any OLE-compliant application. This capability allows applications that use these objects to manage SQL Servers and the databases they contain from remote locations.

### **SQL Executive Service and SQL Server Service**

The SQL Executive Service uses the Transact-SQL language to run scheduled tasks through the command prompt using the SQL Server engine and services. The SQL Server allows you to insert, update and delete data stored in SQL Server.

**An overview  
of ICRISAT  
Genebank**

## **ICRISAT GENE BANK**

The International Crops Research Institute for the Semi Arid-Tropics (ICRISAT)'s genebank was constructed in 1979 with financial support from Japanese government and the Asian development bank, to serve as world repository for the genetic resources of five mandate crops and six small millets

The main objectives of Genebank are:

1. Collection and assembly of Germplasm
2. Maintenance and conservation
3. Characterization and evaluation
4. Distribution
5. Documentation.

The Primary objective of the genebank is to assemble and conserve the endangered landraces and wild species of ICRISAT's mandate crops before their erosion due to destruction of natural habits and replacement by modern high yielding varieties. In this process, ICRISAT Genebank has assembled and conserved over 1,13,000 accessions from 130 countries. The germplasm was assembled by launching collection expeditions in collaboration with NARS in areas of diversity and through donations from various institutions. Collection details including, collectors code, source of sample, date of collection, latitude, longitude and altitude of the collection site, precise location, province and country name and donor related information is maintained in the computer as passport database.

To minimize the regeneration frequency of conserved germplasm, seed longevity is maximized by conserving under cool and dry conditions of the genebank. The germplasm in the genebank assembled will be monitored for seed viability and the quantity at regular intervals. If the viability is < 85% and/or seed quantity is below a critical level, the accession will be regenerated.



The assembled germplasm is characterized for several morphoagronomic traits and evaluated for resistance to biotic and abiotic stresses and the data are stored in computer as characterization and evaluation databases in an easily retrievable form.

The germplasm samples will be supplied free to all bonafied users. Between 1974 and 1998 about 6,39,984 samples were distributed to scientists in more than 130 countries across the world, including India, to ensure unrestricted access. ICRISAT place its collections under the auspices of FAO. The Germplasm designated to FAO is now distributed on a Material Transfer Agreement that stipulates the recipient not to claim ownership or seek intellectual property rights.

#### **SEED CONSERVATION FACILITIES**

The Genebank has three long-term rooms at  $-20^{\circ}\text{C}$  for conserving germplasm for more than 50 years. Six medium term rooms at  $4^{\circ}\text{C}$  and 20-35% RH conserve active collections for 20-25 years. One large short-term storage ( $18^{\circ}\text{C}$  and 35% RH) serves as a temporary holding and processing room of the germplasm.

Active collections under medium-term conditions are stored in aluminum cans with screw caps and rubber gaskets. About 350 g of sorghum, pearl millet, small millets and chickpea, 450 g of pigeonpea and 1 kg. of groundnut pods are stored. Base collections under long-term conditions (longevity > 50 years) are stored in vacuum-sealed aluminum foil packets. About 125 g of sorghum and pearl millet, 300 g of chickpea, pigeonpea and groundnut are stored. All rooms have mobile shelving systems, each capable of accommodating up to 45,000 germplasm accessions. Each room has stand-by refrigeration and dehumidification systems.

The Genebank has a generator to cope with longer periods of power failure.

The Genebank has audio visual alarm system to maintain the desired conditions and smoke sensing and fire fighting devices.

The building that houses the Genebank is protected from earthquake, floods and lightning.

Seed drying facilities operates at 15°C and 15% RH to lower the seed moisture content (5±2%) which is required for long term conservation.

A well-equipped seed lab facilitates to conduct germination tests and seed physiology research on conserved germplasm.

### **Documentation:**

Information plays a significant role in utilization of the germplasm. Accurate information about the conserved germplasm samples is essential if they are to be used well. Computer documentation system enables rapid dissemination of information to users as well as to assist curator in managing the collections more efficiently. The information generated on germplasm collections during day-to-day operations of genebank can be broadly classified into five major categories depending on its source.

1. Passport data (accession identifiers and information recorded by collectors);
2. Characterization and Preliminary evaluation (consists of recording those characters which are highly heritable, can be easily seen by the eye and are expressed in all environments);

Preliminary evaluation (consists of recording a limited number of additional traits through desirable by a management data

3. Distribution (information of germplasm distribution)
4. Management (information indispensable for management of accessions in medium and long-term storage as well as for multiplication/regeneration)

To support the genebank operations and provide easy access to germplasm information, ICRISAT is in the process of developing a genebank management system. It covers the whole range of genebank operations from acquisition of germplasm, through multiplication conservation, regeneration, characterization and distribution to end-users.

# ***System Study***

## **SYSTEM STUDY**

### **REGENERATION AND SEED PROCESSING**

ICRISAT Genebank conserves over 1,13,000 accessions of its mandate crops assembled from different parts of the world. Regeneration of the conserved germplasm is one of the most crucial process in genebank management system as it has a direct bearing on the quality of the material conserved. Germplasm regeneration is a complex process and requires suggestions and ideas to make the process more efficient and cost effective. Processing seed samples for conservation is a critical operation involving several procedural and precautionary measures so as to maintain the genetic integrity of the accessions and to maintain their longevity.

Therefore it is essential to develop a computerized management system to handle information on for regeneration in relation to the seed viability, seed quantity and genetic heterogeneity of each accession. The system should contribute to better planning and implementation of systematic procedures there by assisting the genebank staff in promoting the overall efficiency and cost effectiveness of germplasm regeneration. The module should serve as a decision guide for the genebank curator on regenerating the accession in the collection.

The first step in conservation of genetic resources is assembly of germplasm. This is done by collecting from farmer's fields, particularly in the areas of diversity and security material of interest through correspondence from other plant introduction centers, individuals etc. since germplasm conservation is expensive, acquiring duplicates should be avoided during assembly. However the problem of duplicates in germplasm collections is long pending. Elimination of duplicate accessions based on passport and characterization data some times may be expensive due to loss of valuable genes. However, we can keep the duplicates to a minimum by cross checking the identity, origin etc. of new germplasm accessions with that of existing passport database before registering them. Manual cross checking of above fields is laborious and requires lot of time and energy. Therefore, a suitable software package is required to meet the above requirement.

The proposed module is developed for ICRISAT Genebank Management System (ICGMS), which is dynamic in nature. This module is developed with Visual Basic 6.0 as front end and Microsoft SQL Server 7.0 as back end. The main aim of this module is identification of duplicate Germplasm accessions. This module has following options:

- 1 Import the data**
- 2 Compare the existing database.**
- 3. Print and/or save the list of duplicate accessions.**

The user is allowed to choose any database, which pertains to the Microsoft SQL Server. Every form of this module contains help, which give instructions to users as to what can be done.

Select **IMPORT DATA** option the **SELECT FILE TO IMPORT** dialog box appears. Now, user will select the path (means where the file is located) , file name and the file type (i.e. ...Txt or ...Xls or ..Dbf), then click **OPEN** button in the dialog boxes. This prompts the **SAVE AS** dialog box to appear. In this dialog box user will give the file type of same as in open box or any new name, file name and path (where the user wants to

store) desired and then press **SAVE** button in the dialog box. It saves the file and returns the user back to the main menu for the purpose of comparison.

Selecting **COMPARE EXISTING DATABASES** option, a form appears, with list of tables containing the ICRISAT data and IMPORTED data are displayed in the list boxes. From these tables, select one table from ICRISAT database and one table from IMPORT database the available columns in the table are displayed, select one column in ICRISAT table containing data for comparison and one or more columns as per the requirement in **IMPORT** table with similar data to be compared. Then click **COMPARE** button for finding the duplicates. If duplicates are not found it gives message **THERE ARE NO DUPLICATE RECORDS**, and it asks for repeating the process with another table or not. If duplicates are found, they will be displayed in the grid as a separate output table for viewing.

The view form has **PRINT** and **SAVE** option buttons. On clicking the **PRINT** option button, the duplicate data directly goes to the printer for printing. On clicking the **SAVE** option button, save dialog box appears. Now, enter the file name and path, where the file has to be stored, select the file type then press **SAVE** button in the dialog box. It saves the data either in ...TXT format or ...XLS format depending on choice.

The user can exit by clicking the **EXIT** button in each form. On clicking the **EXIT** button in the duplicates view form, it asks weather you want to repeat the process or not. If the user clicks on **YES** it will go to the main menu form for comparison, if **NO**, it exit the process.

# **System Testing**

# System Testing

## **Testing with sample data:**

Sample data inputs are used to check the system's response to the application. For e.g. the required outputs, their formats etc. There are some problems that can be overcome by testing the sample data. For example the comparison of passport database of one crop is tested with the Import database of the same crop. And also the user for whom it has been developed tests the application for the functionality to trap all the possible errors during its implementation.

Allowing the client to use the system as per the requirements has tested this application and the errors have been trapped. Thus error detection and error handling has been taken care along with the appropriate actions to be taken each time an error has occurred. But in some cases the sample data may not give the desired output. So the sample-input data is only used for testing Basic functionality of the system.

## **Testing with actual data:**

In some cases the actual data is needed to test the functionality of the system. Actual data inputs help the system developer to test all the modules and all the executable statements in each and every module. After the sample-testing phase, one needs to test the system with the actual data by which one can find out any fatal bugs that exist in the system. And, if found, are trapped and the appropriate messages and solutions are provided.

The application has been tested with the client and the errors in all the executable statements in each module have been trapped. Thus the application has been made more user friendly and more successful.



# Software specification

## SOFTWARE SPECIFICATION

**OPERATING SYSTEM : WINDOWS 95**

**FRONT\_END : VISUAL BASIC 6.0**

**DATA BASE : MICROSOFT SQL SERVER**

**UTILITIES : CRYSTAL REPORTS,  
DATA REPORTS,  
MS HELP WORKSHOP,  
MS OFFICE 97**

# **Hardware specification**

# **HARDWARE SPECIFICATION**

<b>PC</b>	<b>:</b>	<b>486 Or Above</b>
<b>RAM</b>	<b>:</b>	<b>32 bit</b>
<b>Disk Space</b>	<b>:</b>	<b>Min 630MB</b>
<b>Operating System</b>	<b>:</b>	<b>Windows 95/NT</b>
<b>Display Type</b>	<b>:</b>	<b>SVGA</b>
<b>Mouse</b>	<b>:</b>	<b>Logitech</b>
<b>Keyboard</b>	<b>:</b>	<b>PC-AT 101Keys</b>

# **Data Dictionary**

**Table: Chickpea Medium Term Active**

**Columns**

<u>Name</u>	<u>Type</u>	<u>Size</u>
ICRISAT Accession Identifier	decimal	9
season Of Harvest	nvarchar	255
Site Of Rejuvenation	nvarchar	255
Date Of Storage	nvarchar	255
Location	nvarchar	255
Seed Quantity(gm)	float	8
100 Seed Weight(gm)	float	8
Seed Moisture(%)	nvarchar	255
Germination(%)	float	8
Date Tested	nvarchar	255
LT_STATUS	nvarchar	255
Remarks	float	8

Table: Chickpea Long Term Base

**Columns**

<b><u>Name</u></b>	<b><u>Type</u></b>	<b><u>Size</u></b>
ICRISAT Accession Identifier	decimal	9
Season Of Harvest	nvarchar	255
Site Of Rejuvenation	nvarchar	255
Date Of Storage	smalldatetime	4
Location	nvarchar	255
Seed Quantity(gm)	float	8
100 Seed Weight(gm)	nvarchar	255
Seed Moisture(%)	float	8
Germination(%)	float	8
Date Tested	smalldatetime	4
MT_STATUS	smalldatetime	4
Duplicate_Status	smalldatetime	4
Remarks	nvarchar	255

Table: Chickpea\_Passport

**Columns**

<u>Name</u>	<u>Type</u>	<u>Size</u>
ICRISAT accession identifier	Number (Double)	8
Accession identifier	Text	50
Table Name	Text	50
Mission code	Text	50
Collector's number	Text	50
Alternate accession identifier	Text	254
Local name	Text	50
Species number	Number (Integer)	2
Cultivar name	Text	80
Biological status	Text	30
Source	Number (Integer)	2
Donor cooperator code	Text	50
Donor geographic code	Text	10
Date received	Text	12
Date of collection	Text	12
Origin geographic code	Text	10
Province	Text	30
Location	Text	254
Latitude	Text	8
Longitude	Text	8
Elevation	Text	8
FAO in trust	Text	1
Remarks	Text	150



Table: Groundnut\_Passport

**Columns**

<u>Name</u>	<u>Type</u>	<u>Size</u>
ICRISAT accession identifier	Number (Long)	4
Accession identifier	Text	50
Table name	Text	50
Mission code	Text	50
Collector's number	Text	50
Alternate accession identifier	Text	254
Field1	Text	50
Local name	Text	50
Species number	Number (Long)	4
Taxon name	Text	100
Botanical variety/forma	Text	50
Cultivar name	Text	80
Biological status	Text	30
Source	Number (Long)	4
Donor cooperator code	Text	50
donor geographic code	Text	8
Date received	Number (Long)	4
Date of collection	Number (Double)	8
Origin geographic code	Text	50
Province	Text	50
Location	Text	254
Latitude	Number (Double)	8
Longitude	Number (Double)	8
elevation	Number (Double)	8
FAO in trust	Text	1
Remarks	Text	255

Table: CHICKPEA (Imported data of chickpea)

**Columns**

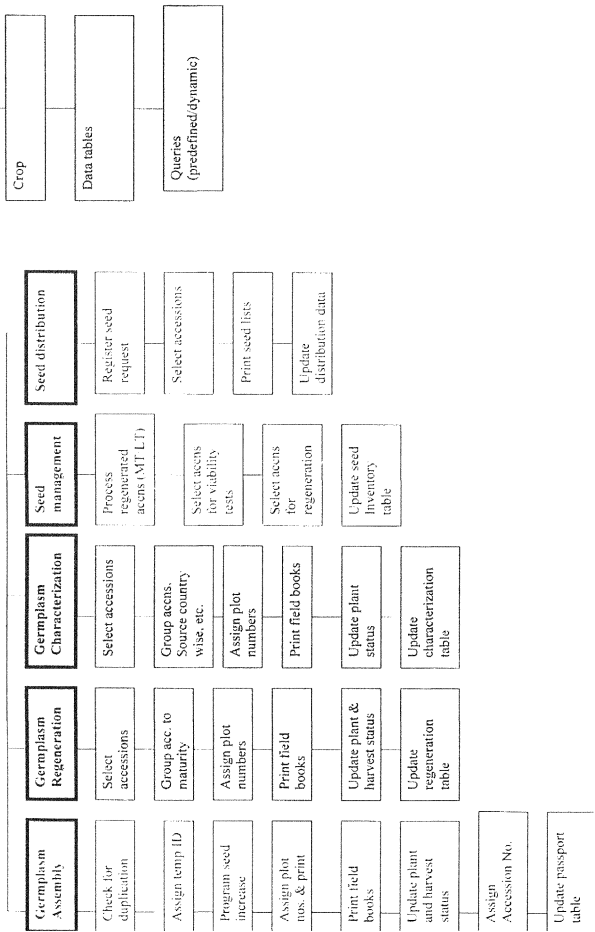
<u>Name</u>	<u>Type</u>	<u>Size</u>
CATALOG	Text	5
REG_UKR	Text	29
NAME_SAMP	Text	25
COU_ORIG	Text	2
ORG_ORIG	Text	3
CHA_ORIG	Number (Double)	8
ACCESS	Number (Double)	8

Table: NPGSPNUT (Imported data of groundnut)

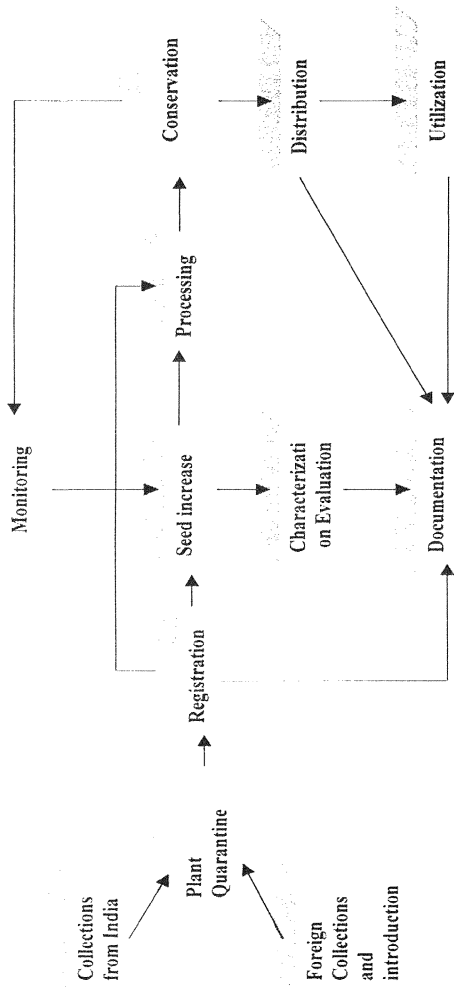
**Columns**

<u>Name</u>	<u>Type</u>	<u>Size</u>
ACID	Number (Double)	8
ACP	Text	4
ACNO	Number (Double)	8
SITE	Text	8
TAXON	Text	42
COUNTRY	Text	20
PLANTID	Text	40

# **Data flow Diagrams**

**Data Management System (Access to limited staff)****Information System (Access to all)**

# Operational Flow Chart of Genebank



**Screens**

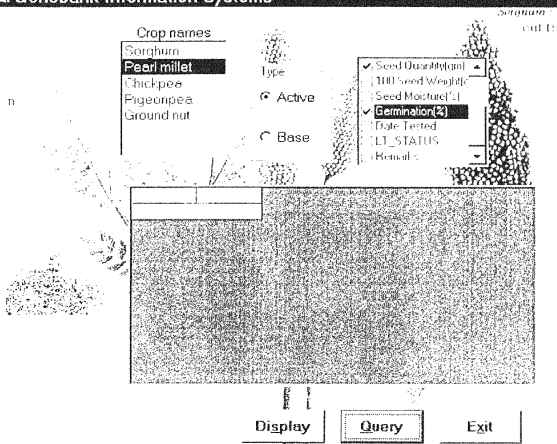
Genbank Information Systems



Regeneration

Database Cross-Referencing

Exit





regeneration



- Crop names
- Sorghum
  - Pearl millet
  - Chickpea**
  - Figeonpea
  - Ground nut

- Type
- Active
  - Base

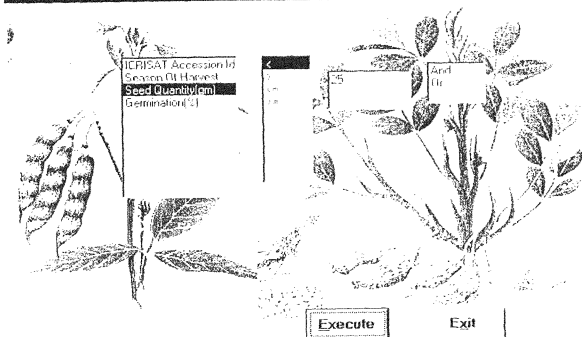
- Seed Quantity(g)
- 100 Seed Weight(g)
- Seed Moisture(%)
- Terminals/ha(1)
- Date Tested
- LT\_STATUS
- Remarks

ICRISAT Access	Season Of Ha	Site Of Rej	Seed Moi	Germination
1	98R			c
2	94R			10
3	94R			10
4	90R			c
5	96R			10
6	91R			10
7	94R			c
8	94R			c
9	94R			10
10	93R			

Display

Query

Exit



ICRISAT Accession Id	Season Of Harvest	Seed Quantity(gm)	Germination(%)
2299	87R	2.64	100
4890	88R	4.8	100
4901	95R	14.29	
7635	76R	21.17	
7656	91R	15.8	
9486	95R	10.02	
14867	89R	13.05	
15554	95R	10.32	
15804	89R	4.02	
16892	88R	23.84	100
16892	97R	3.5	
16954	95R	6.27	
17000	97R	3.22	
17112	97R	24.25	
17112	97R	20.69	

There Are 83 Records

Delete record

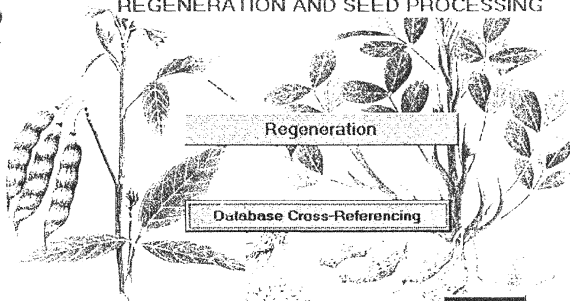
Add record

Print

Save

Exit

## REGENERATION AND SEED PROCESSING



Regeneration

Database Cross-Referencing

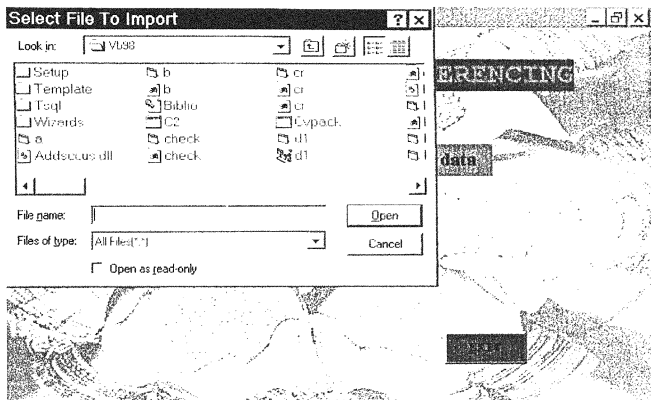
Exit

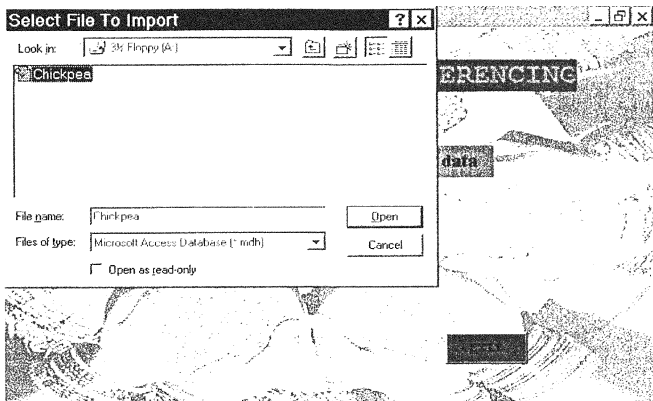
## DATABASES CROSS-REFERENCING

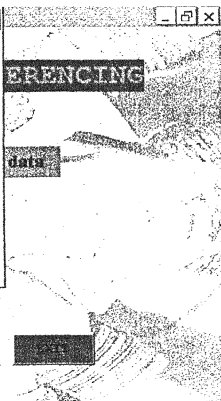
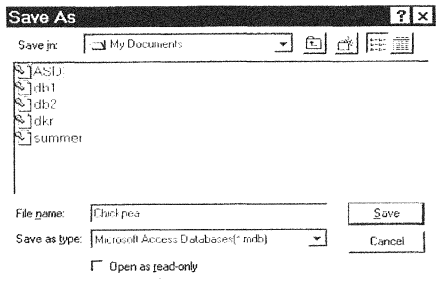
Wish to compare the existing data

I WANT THE DATA

[ ]









## DATABASES CROSS-REFERENCING

Wish to compare the existing data

inventory



The file A:\Chickpea.mdb copied into  
C:\My Documents\Chickpea.mdb successfully

OK

CHICKPEA PASSPORT

Chickpea\_Passport  
Pearl Millet\_Passport  
Groundnut\_Passport

LONG TERM BASE

Long Term Base  
Medium Term Active  
Pearl millet Chara  
al  
as1  
as12  
as41

Complete

Exit

**ICRISAT Table****Chickpea Passport**Pearl Millet Passport  
Groundnut PassportSelect Fields From  
ICRISAT Table  
Chickpea PassportICRISAT accession ▲  
Accession identifi  
Table Name  
Mission code  
Collector's number  
Alternate accessio  
Local name ▼**Imported Table**asdtq ▲  
ASDFR  
**CHICKPEA**  
HPGSPH01  
PERL2  
qzerf  
s1 ▼Select Fields From  
Imported Table  
CHICKPEACATALOG ▲  
 REF\_BKR  
 TRAFI\_SAMP  
 COU\_ORIG  
 ORG\_ORIG  
 CHA\_ORIG ▼**Submit****Print**

Chickpea Passport

Pearl Millet Passport  
Groundnut Passport

ICRISAT Table

ses2  
ses1  
sesht  
sesht1  
table1  
table2  
table3

Select Fields  
ICRISAT Table  
Chickpea\_Passp

inventory  
There Are No Duplicate Records  
OK

Select Fields From  
ted Table  
3

ICRISAT accessio  
Accession ident  
Table Name  
Mission code  
Collector's number  
Alternate accessio  
Local name

ICRISAT Table

ICRISAT Table

<b>Chickpea Passport</b> Pearl Millet Passport Groundnut Passport	sess7 sess1 sess1 sess111 table1 table2 <b>table3</b>
-------------------------------------------------------------------------	-------------------------------------------------------------------------

**Repeat Process** [X]

Wish To Select Another Database

Select Fields  
ICRISAT Table  
Chickpea\_Passp

**ICRISAT access:**  
Accession ident  
Table Name  
Mission code  
Collector's number  
Alternate accessio  
Local name

Fields From  
ed Table

TABLES IN  
CURRENT DATABASE

**Chickpea Passport**  
Pearl Millet Passport  
Groundnut\_Passport

TABLES IN  
EXPORTED DATABASE

- ses7
- ses4
- ses1
- ses11
- table1
- table2
- table3**

Select Fields From  
ICRISAT Table  
Chickpea\_Passport

**ICRISAT accession**  
Accession ident  
Table Name  
Mission code  
Collector's number  
Alternate accessio  
Local name

**Repeat Process** [X]

Wish To Select Another Database

Yes  No

Fields From  
Selected Table

**Compare** **Exit**

Accession Number

Accession Date

Chickpea Passport

Pearl B  
Ground

**inventory**

Enter the table name to store duplicate

OK  
Cancel

Select  
ICRISA  
Chickpe

- ICRISAT accession
- Accession identifi
- Table Name
- Mission code
- Collector's number
- Alternate accessio**
- Local name

fname

duplicate

## Genebank Information Systems

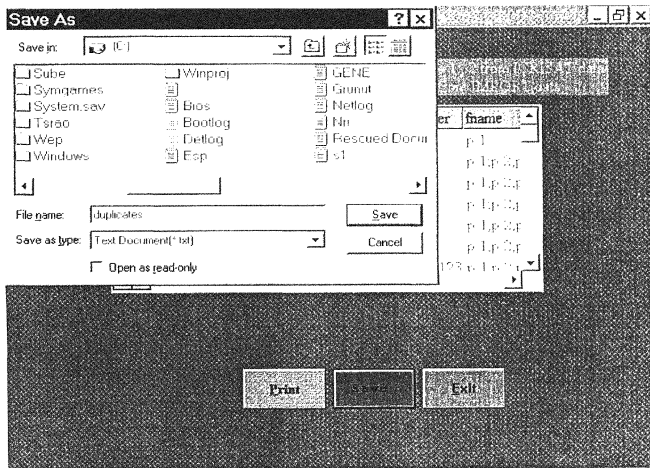
icrisat accession identifier	Alternate accession identifier	filename
1	P 1	p 1
1	P 1	p 1.p 01
2	P 1 1	p 1.p 01
4	P 2	p 1.p 01
18	P 11	p 1.p 01
9532	P 103	p 1.p 01
6422	NFC 410 ET 359694	p 1.p 01

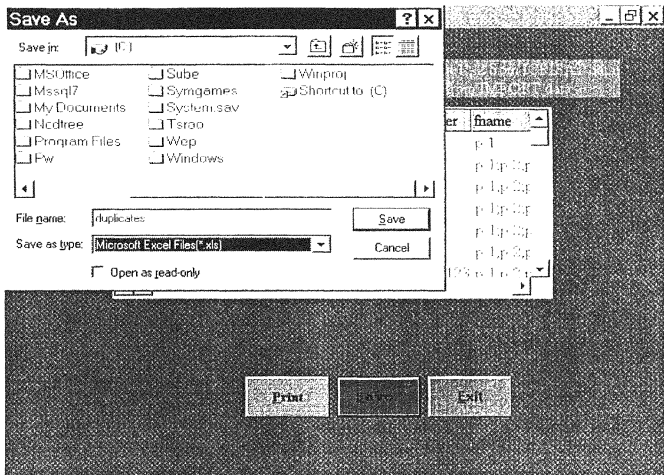
Print

Save

Exit







Microsoft Excel - duplicates

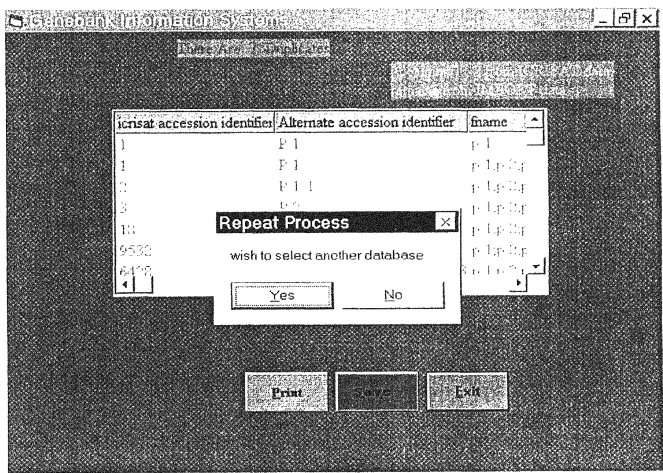
File Edit View Insert Format Tools Data Window Help

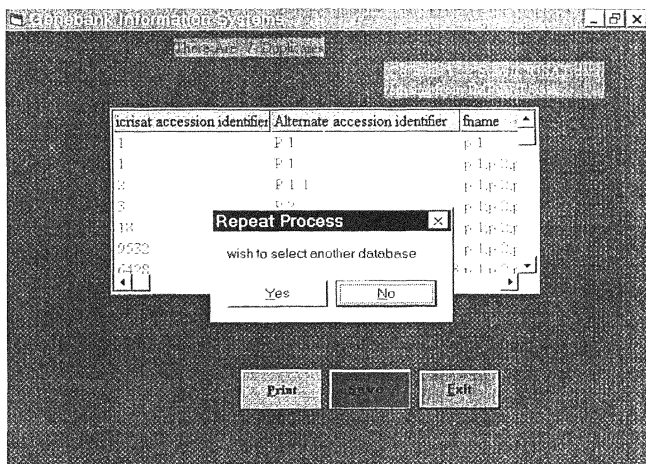
Arial 10 B I U

icnsat accession identifier

	A	B	C	D	E
1	icnsat accession identifier	Alternate accession identifier	iname		
2					
3		1 P 1	p 1		
4		1 P 1	p 1,p 2,p1-1,p123		
5		2 P 1-1	p 1,p 2,p1-1,p123		
6		3 P 2	p 1,p 2,p1-1,p123		
7		18 P 11	p 1,p 2,p1-1,p123		
8		9532 P 123	p 1,p 2,p1-1,p123		
9		6428 NEC 410; PI 359091 ; P 123	p 1,p 2,p1-1,p123		
10					
11					
12					
13					
14					
15					
16					
17					
18					
19					

Sheet1 / Sheet2 / Sheet3





# Reports

Duplicate data save as excel file

icrisat accession identifier	Alternate accession identifier	frame
	1 P 1	p 1
	1 P 1	p 1;p 2;p1-1;p123
	2 P 1-1	p 1;p 2;p1-1;p123
	3 P 2	p 1;p 2;p1-1;p123
	18 P 11	p 1;p 2;p1-1;p123
9532	P 123	p 1;p 2;p1-1;p123
6428	NEC 410; PI 359091 ; P 123	p 1;p 2;p1-1;p123

<b>icrisat accession identifier</b>	<b>Alternate accession identifier</b>	<b>fname</b>
	1 P 1	p 1
	1 P 1	p 1;p 2;p1-1;p123
	2 P 1-1	p 1;p 2;p1-1;p123
	3 P 2	p 1;p 2;p1-1;p123
	18 P 11	p 1;p 2;p1-1;p123
	9532 P 123	p 1;p 2;p1-1;p123
	6428 NEC 410; PI 359091 ; P 123	p 1;p 2;p1-1;p123



# **Bibliography**

## **Bibliography**

<b><u>Author Name</u></b>	<b><u>Title</u></b>	<b><u>Publication</u></b>
1. John Harrington, Eric Winemiller group	Visual Basic 6 Super Bible Vol.1&Vol.2	Techmedia
2. James. A.Senn	System Analysis & Design	Mc.Grawhill
3. Microsoft Corporation	Microsoft Visual Basic Manuals	Microsoft Corporation
4. Microsoft Corporation	Crystal Reports for Visual Basic	Microsoft Corporation
5. Steve Brown	Visual Basic 6	BPB Publications
6. Microsoft Corporation	Microsoft SQL Server Manuals	Microsoft Corporation