INCOME TAX GENERATOR

P- 2280

By

T. RAJALAKSHMI

Reg. No. 71205621032

Of

KUMARAGURU COLLEGE OF TECHNOLOGY, COIMBATORE

A PROJECT REPORT

Submitted to the

FACULTY OF INFORMATION AND COMMUNICATION ENGINEERING

In partial fulfillment of the requirements

for the award of the degree

of

MASTER OF COMPUTER APPLICATION

June 2008

BONAFIDE CERTIFICATE

Certified that this project report titled **"INCOME TAX GENERATOR"** is the bonafide work of **Ms. RAJALAKSHMI. T (Register No: 71205621032)** who carried out the research under my supervision. certified further, that to the best of my knowledge the work reported herein does not form part of any other project report or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

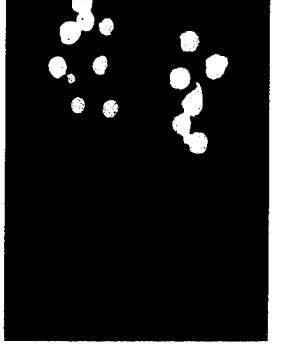
Project Guide

Head of the Department

Submitted to Project and Viva Examination held on ----- 1-07-2008 -----

Internal Examiner

External Examiner



PROJECT COMPLETION CERITIFICATE

This is to certify that the project report entitled **INCOME TAX GENERATOR** that is being submitted by **Ms.Rajalaksmi.T (Register No: 71205621032)** in partial fulfillment for the award of the Degree of Master in Computer Application (MCA) in **Kumaraguru College of Technology** is a record of bonafide work carried out by her under my guidance and supervision. The results embodied in this project report have not been submitted to any other University or Institute for the award of any degree or diploma.

Date: 12 June 2008

Signature: J. Dem

Designation: PROJECT MANAGER - R&D SERVICES.

ABSTRACT

A decade ago, there were very few people who pay income tax to the government, but now due to the entry of multi national companies, the salary level has been hiked, and the number of people who pay tax have also been increased. Each individual has to afford tax to the government if their salary is more than a certain amount. Earlier, it was the responsibility of the individual to pay tax to the government, but now the rules have been changed. The employer has to deduct a certain amount every month from the salary of the employee, when the yearly salary exceeds the tax reduction limit. This amount is called tax deduction at source (TDS). The company pays the TDS amount to the government every month.

Every organization wants to help their employees in tax deduction and even the organization wants to reduce the expenses paid for the chartered accountants for calculating their employees' tax and TDS amounts. An effective solution for the above is to build a system for managing all the calculations and maintenance operations.

The project entitled 'Income Tax generator' is a system, which manages all the income tax related operations. The system educates the employees on the tax terms and tax saving methods. The system calculates the income tax for the employees and their monthly TDS amount. It also facilitates the employee to enter their saving and loan details, which contribute the tax reduction. It also provides information, about various ways in which the tax can be saved. The system itself applies all the rules and slabs for the employees' salary and calculates tax. The system also sends a report automatically to the finance department of the company, which contains all the information and calculated amount of TDS that has to be deposited in the Government's account.

The system is designed and developed using J2EE technology. JSP for designing the user-interface, Servlets to control the actions and Core Java for the business logics. MySQL is used as back-end.

ACKNOWLEDGEMENT

I wish to express my sincere gratitude and heartfelt thanks to our beloved Principal **Dr.Joseph V. Thanikal**, Kumaraguru College of Technology, Coimbatore for permitting me to undertake this project.

I express my gratitude to **Dr. M. Gururajan**. Head of the Department of Computer Applications for extending his support throughout the project by all means and his continuous motivation which made me to complete the project successfully.

I extend my heartfelt thanks and gratitude to **Mrs. V. Geetha**, Assistant Professor, Department of Computer Applications, for rendering me all the timely help and valuable guidance throughout the project.

I also extend my gratitude to the Project Mentors **Mr. Noufal Karaparambath** and **Mr. Jesvin Maharoot**, MindTree Limited, Chennai for their careful supervision that has ensured me attaining perfection of work.

I also thank my parents, friends and everyone who was with me supporting throughout the project.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 PROJECT OVERVIEW

The project entitled "Income Tax Generator" is designed and developed using J2EE technology. The System has been designed by using JSP and Servlet Technology. The system design follows the Model, View and Control (MVC) Architecture for development. MySQL is supporting the entire System as a power back-end.

The existing system exists but it is a simple manual application. It supports only manual entry and maintenance of records. The process of calculating tax and tax deduction at source (TDS) is a tedious and time-consuming process. The proposed system is developed to over come all the gaps between the employee salary record maintenance and the tax calculation process.

The project is divided into six modules

- Authentication Module.
- Employee Module.
- Tax Calculation Module.
- Savings Module.
- Maintenance Module.
- Report generation Module.

1.1.1 Authentication Module

Authentication module covers the entire user authentication processes, to enter into the website. Since the system developed is a web-oriented system, security places a major role. The

employee-id of each user is used as username. The user session in the website is maintained. User level-restrictions are done in this module. Password and username are encrypted and transferred using SSL (Socket Secure Layer). The Username and password is encrypted and stored in a database.

1.1.2 Employee Module

This module is used to store all the employee details. This includes employee personal details and all the details needed to calculate income tax such as salary, HRA, DA, TA, loans etc. All the salary details come from the payroll department. All the employee relevant tasks like updating, modification, deletion, etc. are kept track here.

1.1.3 Tax Calculation Module

This module contains all the details needed for calculating tax. The user income and other savings, loans are taken in account and calculation of TDS is done. The income tax is then calculated by applying all the advanced tax rules and latest income slabs. The system verifies all the details of the employees. The system generates a template for submitting the income details to the IT department.

1.1.4 Savings Module

This module is used to maintain all the employee savings. There are various sections under which an employee can make savings and can claim for tax reduction. So sections and saving policies approved by the company are maintained here. When an employee enters his/her savings, the system check for the policy's upper limit as well as the sections upper limit and this module helps in validating it.

1.1.5 Maintenance Module

In this module all the tax rules and slabs for various categories of employees are maintained, since rules and slabs are subject to change with Government norms, there will be a change in the policies and rules every year. Only the administrator is allowed to make change in the rules and slabs. Administrator authentication is done here. Any updating in the system is done through this module.

The employees declare their savings in the beginning of every financial year. The administrator has to check proof and validate the savings done by the employees. The accountant updates the status of each employee.

1.1.6 Report Generation Module

This module helps in getting the present and past records of the employees, which may be needed by the higher authorities or by other income tax officers. It generates all the reports based on date, employee wise, year, list of employees whose tax amount is above or below or within a set of range, etc.

1.2 ORGANIZATIONAL PROFILE

MindTree Limited was started in 1999 by a diverse team of 10 professionals who came from three different nations and had already scripted successful careers. Their vision to build an institution that is among the most admired companies globally is shared with every MindTree Mind and is reflected in the way we do business.

Mission

The mission statement of MindTree reflects the customer backward approach and relentless focus on company's people, the MindTree Minds. The mission of MindTree is to: **"Deliver business-enabling solutions and technologies by creating partnerships with our customers in a joyous environment for our people."**

MindTree is among the fastest growing IT and R&D services company. The growth of the company has been powered by its consultative approach, their deep understanding of business and technology, their passion for innovation - and above all, its integrity.

Consistent performance has been the hallmark of the company's success. Its focus on innovation ensures that the company is constantly raising the bar for itself and for the industry as a whole. MindTree takes a customer-backwards view to every one of its engagements, focusing on value delivered as the benchmark for its achievements. MindTree is proud to be one of the few Indian companies to have maintained a consistently high customer satisfaction rating over the years.

IT Services

An efficient IT solutions strategy helps organizations out-pace competition and foster innovation. MindTree's comprehensive range of end-to-end IT services is designed to satisfy the needs of businesses of every size to compete and succeed in the global market.

MindTree has helped global organizations gain competitive edge by leveraging the right mix of technology, people, and processes to achieve strategic objectives. Our team of domain experts ensures that IT initiatives are tied to business imperatives through quantifiable metrics. Some of the services are

- Application Maintenance Services
- Business Intelligence
- Mainframe and System
- Infrastructure Management and Technical Support
- Internet and Emerging Technologies

R&D Services

Market dynamics have made it imperative for technology companies to innovate and roll out products at a faster pace and at lower costs. Partnering with the right company to build products correctly the first time and co-innovate in new technology is critical to success. Mind Tree's domain-specific end-to-end R&D services are designed to help organizations meet their engineering needs, enabling them to succeed in the global market.

MindTree's technology building blocks, in the form of Intellectual Property and frameworks help bridge technology gaps and bring products to market faster. Our product-centric services approach and mature ecosystem partners allow us to provide value beyond engineering services through complete product ownership. Some of the services are

- IP Led Services
- Semiconductor and Hardware Engineering
- Software Engineering
- Testing

CHAPTER 2

SYSTEM ANALYSIS

System analysis refers to taking known facts concerning a system, breaking these into their elements and establishing logical relationship between the elements, with the objective of producing a specification of requirements.

2.1 EXISTING SYSTEM

The various aspects of the existing system are thoroughly analyzed and the need for the proposed system is taken into account. In general, the existing system follows the manual approach. Using this manual system, the users can file their savings, loans and other declarations through an email or through printed copies. The accountant in charge for calculating the income tax has to go through either all the emails or through printouts.

The Accountant has to file the copy of the proof for savings, loans very carefully. These filed copies are sent to the chartered accountants to calculate the income tax. The chartered accountant calculates the tax for the employees. This process is carried down at the beginning of every month. The calculated tax deduction at source amount is then sent to the payroll department for calculating the monthly salary of the employee. The deducted TDS amounts of all the employees are sent to the finance department so that the TDS will be deposited to the Government account.

The existing system never ensures a demonstration process to the employees how the tax is calculated and under which criteria the tax is deducted at every year. This results in employee dissatisfaction.

The problems in the existing system are,

- Time consuming.
- Very expensive.
- Less secure since records are maintained in file.
- Less efficient in accessing data.
- Consumes lot of man power.
- Sometimes leads to inaccuracy in calculations of tax and miscalculation may happen.

2.2 PROPOSED SYSTEM

The Income Tax Generator is a complete variation of the manual calculating system. The developed system is incorporated into the company's internal website. The employees only have to log in and can view information about various dimensions used in calculating tax. The information about the various Government norms and procedures are provided in the website. The employee can view and understand the tax terms.

The employees can file their savings and apply for deductions that are possible for tax deductions approved by the government and the company. The employee can get access to all yearly/monthly salary, TDS and all other tax details. The tax calculation also is made easier with the automated tax calculator.

The accountant's job is made easier. The amount of TDS deducted for all the employees are sent automatically to the finance department for deposition of it in Government account. The search of earlier records is very efficient and simple with the used efficient search techniques.

The benefits of the proposed system are,

- Efficient when compared to the manual system.
- Record maintenance is made easy and simple.
- Reduces manpower consumption.
- Inexpensive and less time consumption.
- User satisfaction and informative.

2.3 PROBLEM FORMULATION

The main objective of the Income Tax Generator System is to computerize the manual work of computing income tax and tax deduction at source (TDS). It wants the employees of the organization to access their IT information through web. The system is going to operate in the World Wide Web and used by the employees of the organization.

In the financial department of the company all the calculations for submitting the IT statements are done for each employee. For each employee there are various IT deductions and slabs. The slabs and advanced tax rules change in accordance with the Government norms.

The system should be flexible for updating the reviewed IT slabs and rules whenever the Government makes the changes. It should also store relevant details about each employee and the savings done by the employee since it might affect the TDS amount. The system should also calculate the TDS for each employee and should send a report to the finance department periodically. The system should facilitate the employee to feed the information about their savings themselves.

2.4 FEASIBILITY ANALYSIS

Feasibility analysis is the measure of how beneficial or practical the development of an information system will be to the organization. Once the problem is explained, information is gathered about the system to test whether the system is viable technically, financially and operationally.

2.4.1 Feasibility Considerations

The key considerations that are involved in the feasibility analysis are:

- Economic
- Technical
- Operational

2.4.1.1 Economic Feasibility

Economic feasibility is the measure of the cost-effectiveness of the proposed system. The investment to be made in the proposed system must prove a good investment to the organization by returning benefits equal to or exceeding the cost incurred in developing the system.

The proposed benefits of the system outweigh the cost to be incurred during the system development. The system does not require any special hardware facilities since the system is going to be implemented in the company's internal website, it is economically feasible. It also reduces the outsourcings cost that is being spent for calculating TDS to the chartered accountants. It is only one time development expense. In addition capability of the system to incorporate future enhancements will improve the performance to suit the future needs of the user and the fact that a single system can be used for a standalone organization as well as a corporate organizations prospect.

2.4.1.2 Technical Feasibility

Technical Feasibility is the measure of the measure of practicality of a specific technical solution and the availability of technical resources and expertise. It centers on the existing computer system (hardware, software, etc.) and to what extent it can support the new addition.

The system is developed using J2EE and MySQL as back end. The GUI will be developed using JSP and JDBC for persistence. The business logic will be implemented using core Java. The tools used for this project are:

- Star UML for use case diagrams
- Struts framework for the web

These resources are already available with the organization along with the hardware resources that might be needed for the proposed system. Hence technically the system is feasible.

2.4.1.3 Operational Feasibility

The resources that are required to implement are already with the organization. The personnel of the organization already have exposure to computers. So the project is operationally feasible. The proposed system has found encouraging support from the management as it will be great use to them. The employees of the organization are also committed to have the system operational as it will save time and reduce their workload.

CHAPTER 3

SYSTEM REQUIREMENTS

The Software Requirement Specification is a technical specification of requirements for the software product. The goal of software requirements definition is to completely and consistently specify the technical requirements for the software products in a concise and unambiguous manner. The software requirements specification is based on the system definition and high-level requirements that are specified during initial planning of the project. It is an elaborated and more specific, in order to characterize the features that the software product will incorporate.

The requirement specification is primarily concerned with functional and non-functional performance aspects of the software product and the emphasis is placed on specifying the product characteristics without implying how the product will provide those characteristics.

3.1 HARDWARE REQUIREMENTS

- Processor : Intel Pentium /AMD
- Hard Disk : 80GB
- Display : VGA
- Floppy disk : 1.44 MB
- RAM : 512 MB

3.2 SOFTWARE REQUIREMENTS

- Operating system : Windows XP/ Windows Vista.
- J2EE Framework : JSP, Servlet, JDBC, HTML.
- Database : My SQL.
- Web Server : Apache Tomcat 5.0

3.3 SOFTWARE SPECIFICATION

3.3.1 J2EE

J2EE is one of the most popular editions of Java in the market today. Sun Microsystems has positioned it to fit multiple purposes in development. The J2EE environment ranges from core java to the most advanced forms of enterprise java technologies like Servlets, JSP, EJB, Beans, JMS, JDBC, JNI and web services etc.

We can easily develop applications and enhance them using J2EE. J2EE ranges from lightweight JavaScript programming, to application-specific programming with J2EE for application and finally full fledged enterprise development.

Features:

- Visual Platform
- Ease of Use
- Flexibility
- Ease of Enhancement
- Easy to Understand
- Efficient
- Powerful
- Portable

Efficient

With traditional CGI, a new process is started for each HTTP request. If the CGI program does a relatively fast operation, the overhead of starting the process can dominate the execution time. With servlets, the Java Virtual Machine stays up, and each request is handled by a lightweight Java thread, not a heavyweight operating system process. Similarly, in traditional CGI, if there are *N* simultaneous request to the same CGI program, then the code for the CGI program is loaded into memory N times. With servlets, however, there are *N* threads but only a single copy of the servlet class.

Powerful

Java servlets let you easily do several things that are difficult or impossible with regular CGI. For one thing, servlets can talk directly to the Web server (regular CGI programs can't). This simplifies operations that need to look up images and other data stored in standard places. Servlets can also share data among each other, making useful things like database connection pools easy to implement. They can also maintain information from request to request, simplifying things like session tracking and caching of previous computations.

Servlets

Java Servlets are more efficient, easier to use, more powerful, more portable, and cheaper than traditional CGI and than many alternative CGI-like technologies. (More importantly, servlet developers get paid more than Perl programmers).

Java Server Pages

Choosing a server-side language used to be easy. Way back when, CGI was pretty much the only scripting option out there. Intrepid developers could write their own server extensions, but few were up to the challenge, and the gap between these two options (inefficient CGI scripts and extremely complex server extension development) was huge.

Microsoft tried to fill this hole with Active Server Pages (ASP), which allow developers to use simple scripting to access the server and its extensions. But while ASP gives you an efficient way to return dynamic content, it essentially limits you to Microsoft platforms, and even the simplest of scripting mistakes can cause the server to crash or hang, effectively bringing down your website.

JSP provides us with access objects that we can use to get information from the incoming request and modify the resulting response. This makes it possible for us to do things like identify what type of browser is making the request. We can then set up a different type of request based upon what that browser supports

Eclipse

Eclipse is an extensible, open source IDE (integrated development environment). The project was originally launched in November 2001, when IBM donated $40 million worth of source code from Web sphere Studio Workbench and formed the Eclipse Consortium to manage the continued development of the tool.

The stated goals of Eclipse are "to develop a robust, full-featured, commercial-quality industry platform for the development of highly integrated tools." To that end, the Eclipse Consortium has been focused on three major projects:

MySQL

MySQL is a database. A database is a data storage feature. It can be used to store, sort, arrange, and display information. MySQL is a functional feature on its own. For our tutorials, we will be using PHP commands to use the functions of a MySQL database. Some hosts may have this program available. MySQL is a data storage area. In this storage area, there are small sections called tables. Very similar to a normal HTML table, the MySQL tables consist of rows, columns, and cells.

3.4 SYSTEM FUNCTIONAL REQUIREMENTS

The following are the functional requirements of the system.

- The system should provide proper authentication to the users of the system.
- The system is web oriented, so the users of the system should be restricted to the employees of the company alone.
- Once an employee leaves the company he should not be able to use the system.
- The slabs for paying IT for men, women and the senior citizen vary according to the government norms must be maintained. Facility for updating these should be provided.
- Permission for updating the system properties and function should be given only to the administrator. But an employee can update or change his data any time.
- The system should calculate TDS for each employee.
- The system should maintain monthly gross pay and TDS amount of each employee, since increment in salary may affect the calculation of TDS.
- The amount of savings that the employee is going to do for the current financial year should be obtained from the employee and should be stored in the database.
- The approval of documents for the pre-quoted savings should be done at the month of june. So the status of each employee should be maintained.
- The system should provide a template (form 16) for submitting the IT statement to the finance department, so that the employee can fill in it and submit to the accountant manually.

3.5 NON-FUNCTIONAL REQUIREMENTS

- **Availability**

The system should be available for all the users. It mainly depends on the availability of the server.

- **Scalability**

The system should support minimum of 100 users at a time. The speed should be reliable.

- **Maintainability**

The system can meet changes very easily. The requirements of the software while evolving will be met by just adding new sub-functions. The only major changes in the system would be the change in IT slabs and rules. These details are stored only in an XML file so updating it is not a major issue. Therefore, the maintainability of Income Tax System would not be a complex issue.

- **Security**

The system has an authorization method for allowing users in to the system. Each and every user of the system is authenticated by verifying passwords. It will be using 128-bit SSL secure connection over the web, thus guaranteeing security.

- **Documentation and user manual**

The complete working of the system should be documented and should be given to the user. The functionality of the system should be delivered to the user. All the documents and user manual should be user friendly.

CHAPTER 4

SYSTEM DESIGN

System design refers to an abstract representation of the system. It concerned with making sure the system will meet the requirements of the product, as well as ensuring the future requirements can be addressed. It also addresses interface between the system and other products. Major step in system design is the preparation of the input forms and the output reports in a form applicable to the user.

The main objective of the system design is to use the package easily by any computer operator. System design is the creative act of invention, developing new inputs, a database, offline files, method, procedures and output for processing business to meet an organization objective. System design builds information gathered during the system analysis.

4.1 ARCHITECTURE DESIGN

4.1.1 Architectural Decisions

The tools used for this project are:

- Star UML for use case diagrams
- Eclipse for implementing.

This application is not required to take high load as the maximum users expected to be logged in simultaneously is 100 users. The client does not require a very fast system but is particular on ease of use as his staff is not tech-savvy. Hence the design of the website is simple without too many navigations and options.

4.1.2 Layering architecture

The system consists of four layers, namely

- Client.
- Listeners, Application Controllers and Business Event Translators
- Business Layer, Data Access Layer and Adapters.
- Enterprise Database Layer.

The architectural layers of the system are depicted in Figure 4.1.2.1.

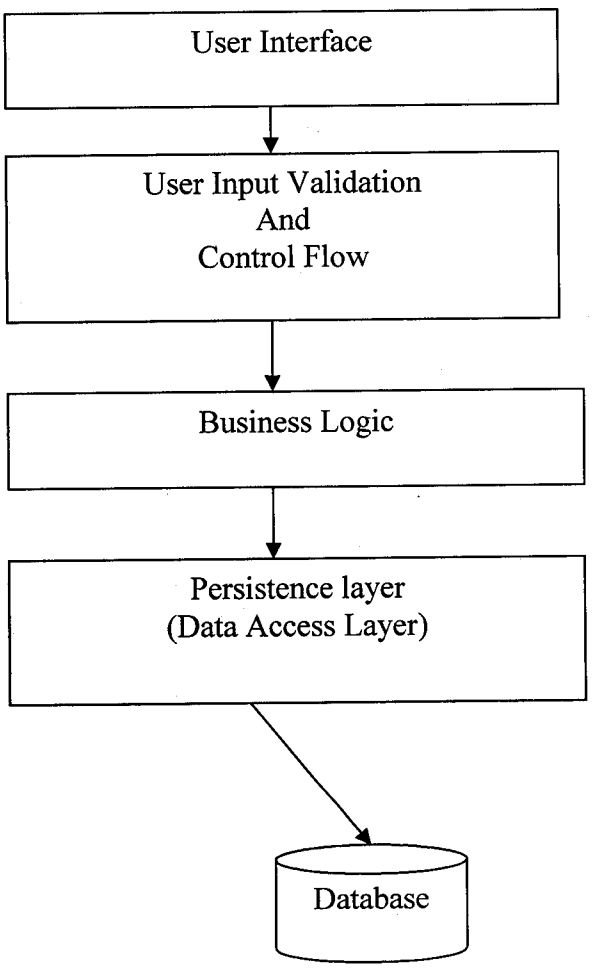


Figure 4.1.2.1 Architectural layers

The user interface will consist of a web interface that will display to the user the details of employee, slabs, rules and also allow users to add/update slabs and advanced tax rules details. The details that users enter will go through a through validation before being presented to the logical layer. This is a separate layer as the validations can be done by data types (example, dates, numbers, strings etc) and these common validations can be used throughout the system.

The business logic layers takes care of the functionalities required of the system, namely searching for available employee details, updating details of advanced tax rules, updating status of submission of IT statement so that it does not need to be generated for the same employee again. The persistence layer involves connecting to the database and retrieving and updating into it.

4.2 PROCESS MODELS

The software process model maybe defined as a simplified description of a software process, presented from a particular perspective. In essence, each stage of the software process is identified and a model is then employed to represent the inherent activities associated within that stage. Consequently, a collection of local models may be utilised in generating the global picture representative of the software process. Examples of models include the **workflow model**, the **data-flow model,** and the **role model.**

- The **workflow model** shows the sequence of activities in the process along with their inputs, outputs and dependencies. The activities in the model represent human actions.

- The **dataflow model** represents the process as a set of activities each of which carries out some data transformation. It shows how the input to the process such as specification is transformed to an output such as design. The activities here maybe lower than in a workflow model. They may represent transformations carries out by people or computers.

- The **role model** represents the roles of people involved in the software process and the activities for which they are responsible.

Unified Modelling Language

The UML is applicable to object-oriented problem solving. Anyone interested in learning UML must be familiar with the underlying tenet of object-oriented problem solving -- it all begins with the construction of a model. A model is an abstraction of the underlying problem. The domain is the actual world from which the problem comes.

UML models may be automatically transformed to other representations (e.g. Java) by means of QVT-like transformation languages, supported by the OMG. UML is extensible, offering the following mechanisms for customization: profiles and stereotype.

The Income Tax Generator uses the UML for designing since it is built using Java. The representations used in UML diagrams can be easily related and converted to Java.

UML uses various diagrams for designing a system. In this project, use case, sequence, collaboration and class diagrams are used.

4.2.1 USE CASE DIAGRAM

The use case diagrams are used to show various users roles, who are all the actors and what actions they perform.

The requirements are represented using the use case diagram and detailed use case specifications.

Actors:

Following actors will use the system:

1) Accountant (User)
2) Employee (User)

Both the actors will be able to login, logout and file their income tax details. The administrator should be able to update slabs and advanced rules.

Actor Goals:

The Figure 4.2.1.1 and Figure 4.2.1.2 illustrates the goals of each actor.

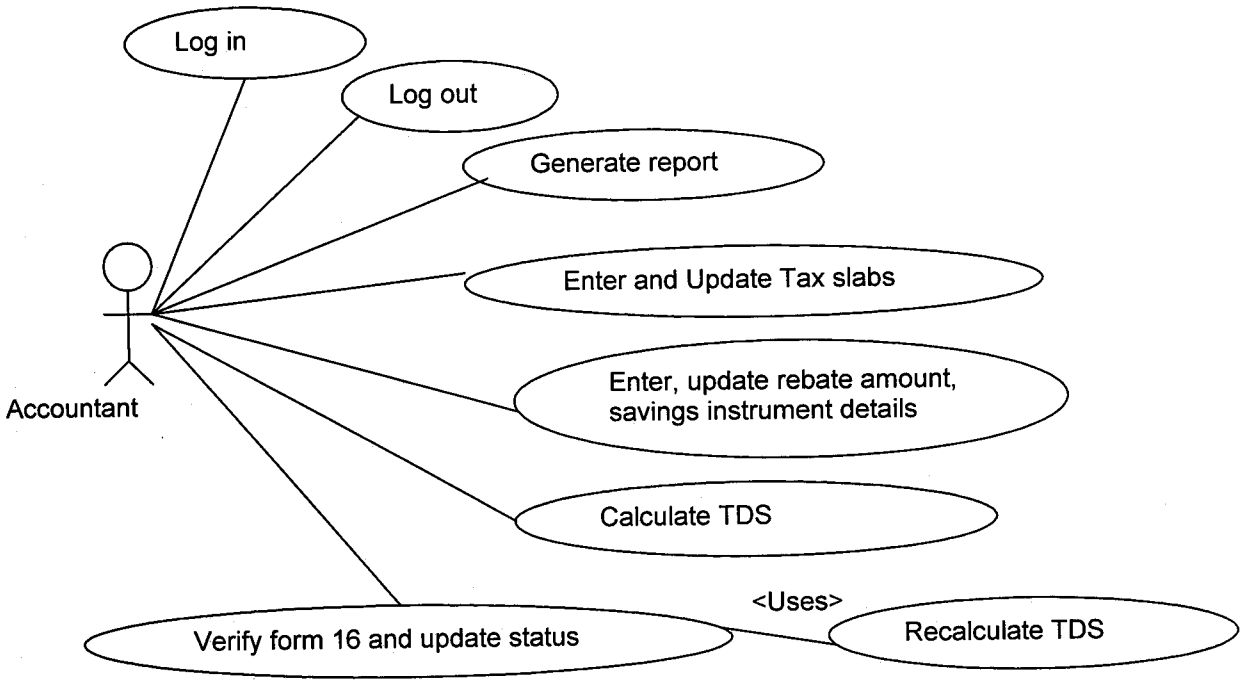


Figure 4.2.1.1 Use Case Diagram for an Accountant

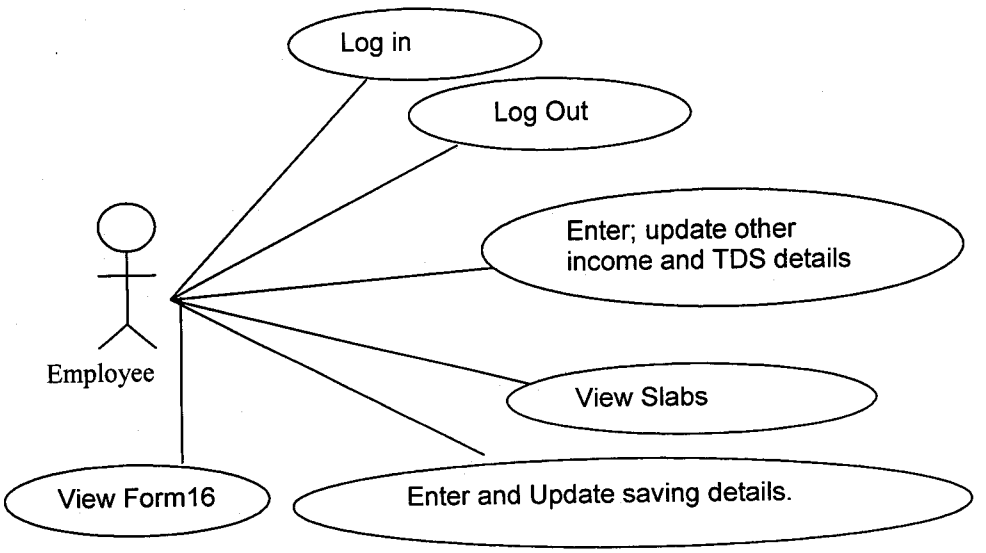


Figure 4.2.1.2 Use Case Diagram for an Employee

Actions:

The various action that take place in the system are shown in the below use case diagram.

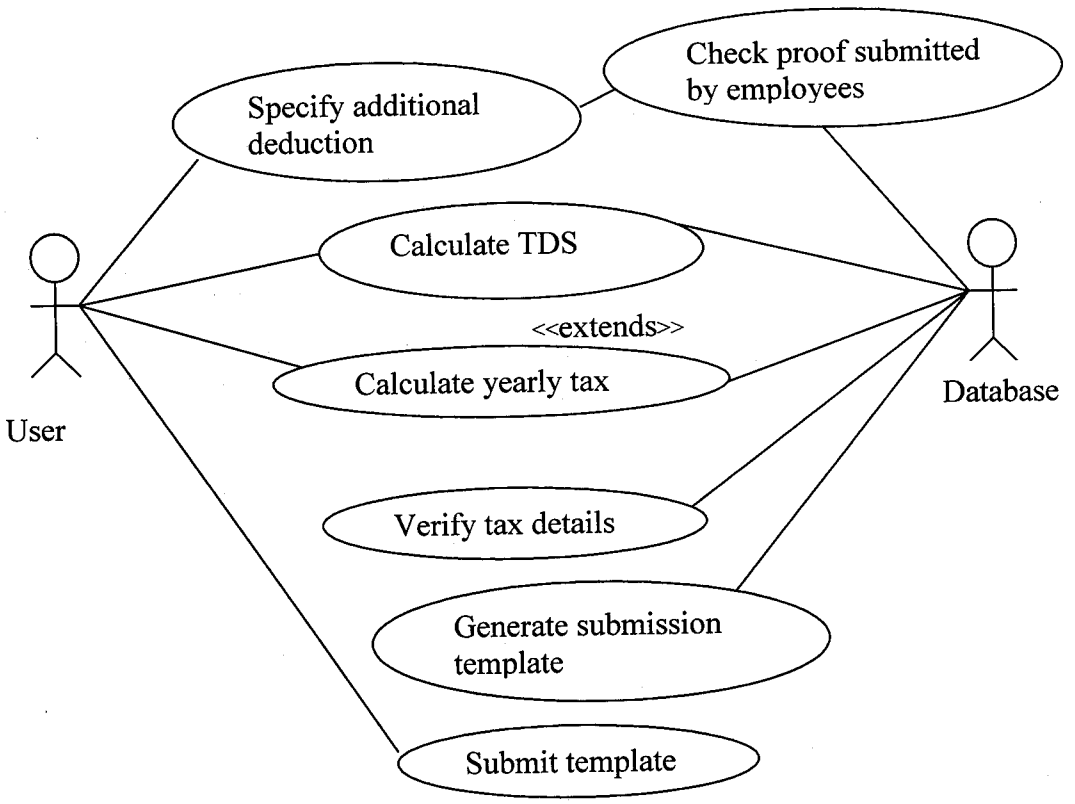


Figure 4.2.1.4 Use Case Diagram for Tax Calculation Module

4.2.2 USE CASES DESCRIPTION

The use case description are shown from Table 4.2.2.1 to Table 4.2.2.6

USE CASE 1	**Login**
Goal in Context	User should login to the company website
Preconditions	User knows how to get to the website
Success End Condition	Successful login
Failed End Condition	Login fails
Actors	Employee, Accountant
DESCRIPTION	
User	System
User keys in user name and password	System validates username and password by verifying it in the Master database. The username is the employee id of the user. It is retrieved from the master database. System displays welcome screen and options.
Alternate Scenario	
User	**System**
User keys in name and password	• System validates name and password • System displays error message and gives back the login page again

Table 4.2.2.1 Use case – Login

USE CASE 2	**Manage Savings Details**
Goal in Context	User should be able to update savings details information
Preconditions	User is already logged in
Success End Condition	Successful update
Failed End Condition	Updating may fail due to one of many reasons
Actors	Employee
DESCRIPTION	
User	**System**
User chooses the update option from the website User enters the policy id and rebate amount and the section of the rebate and clicks update	• System displays information page that allows the user to Enter/update his/her savings details and the section under which the savings is done. • The system checks the savings amount in the Savings_details database and if it is less than the maximum amount, it updates the details in the Employee_Savings database
Alternate Scenario	
User	**System**
User chooses the update option from the website User enters the details and selects the update option.	System displays the information page that allows the user to update his/her details. System displays an error message due to some connection or database problem.
User chooses the update option from the website User enters the details and selects the update option.	System displays the information page that allows the user to update his/her details. System displays an error message since users total savings amount exceeds the maximum amount.

Table 4.2.2.2 Use case – Manage Savings Details

USE CASE 3	**Manage saving policy and section details**
Goal in Context	The various savings policy accepted by the company to show deduction should be entered and updated successfully.
Preconditions	Accountant is already logged in
Success End Condition	Successful updating
Failed End Condition	System error
Actors	Accountant
DESCRIPTION	
User	**System**
Accountant selects the update savings policy option.	System displays the updating page with already existing savings policy if any and its corresponding deduction percentage amount and section.
The Accountant enters the new policy and its corresponding deduction percentage.	The System updates the value in the corresponding Savings_Instrument database and displays a message as "updated"
Alternate Scenario	
User	**System**
Accountant selects the update savings and deduction percentage option.	The system display error message due to some problem like failure of database server or connection. The system may display error since the input details may not be appropriate.

Table 4.2.2.3 Use case – Manage saving policy and section details

USE CASE 4	**Generate Report**
Goal in Context	The required report should be generated.
Preconditions	User is already logged in
Success End Condition	Successful report generation
Failed End Condition	System error
Actors	Accountant
DESCRIPTION	
User	**System**
The accountant selects the generate report option from the website.	System displays various options like employee-wise report, persons without tax, range-wise reports.
The accountant selects one of the options and specifies employee id for employee-wise report, range for range-wise report	The System inputs the user choice and generates report based on it.
Alternate Scenario	
User	**System**
The accountant selects the generate report option from the website.	The system display error message due to some problem like failure of database server or connection.
The accountant selects one of the options.	The system may display error since the option may be invalid.

Table 4.2.2.4 Use case – Generate report

USE CASE 5	**Verify proofs and update status**
Goal in Context	The status should be updated in the Employee_Tax_Rebate database.
Preconditions	User is already logged in
Success End Condition	Successful verification and updating status.
Failed End Condition	System error
Actors	Accountant
DESCRIPTION	
User	**System**
The Accountant selects the update status option from the website.	System displays the update status page to the user. System displays the various options for entering the employee_id and a list of status applicable.
The Accountant enters the employee id and the status as verified, disapproved, rejected	The system validates the employee id. It updates the status in the Employee_Tax_Rebate database for the employee.
Alternate Scenario	
User	**System**
The Accountant selects the update status option from the website	System displays the update status page to the user. System displays the various options for entering the employee_id and a list of status applicable.
The Accountant enters the employee id and the status as rejected	The system validates the employee id and sets status as rejected.
The Accountant selects the recalculate TDS option	The System recalculates the TDS amount for the yearly gross salary excluding the savings deduction and updates the current month TDS with the remaining payable amount.

Table 4.2.2.5 Use case – Verify proofs and updates status

USE CASE 6	**Calculate TDS**
Goal in Context	The TDS amount of the user should be calculated
Preconditions	User is already logged in
Success End Condition	Successful Calculation
Failed End Condition	System error
Actors	Accountant
DESCRIPTION	
User	**System**
Accountant selects the Calculate TDS option from the website.	System displays the TDS calculation page showing various options like TDS for all Employees or a particular employee.
The user selects the "all employee" option	The System retrieves the monthly salary of each employee from the Payroll database and calculates TDS amount. And stores it in the Employee_TDS database.
The user selects the single employee option and enters the employee_id.	The System retrieves the monthly salary of that employee from the Payroll database and calculates TDS amount. And stores it in the Employee_TDS database and displays the amount
Alternate Scenario	
User	**System**
The user selects the Calculate TDS option from the website	The system display error message due to some problem like failure of database server or connection.

Table 4.2.2.6 Use case – Calculate TDS

4.2.3 SEQUENCE DIAGRAM

Sequence Diagrams are used to depict the sequence of actions that occur in the system. It represents the Dynamic behavior of the system. The Figure 4.2.3.1 depicts the sequence diagram for calculating tax.

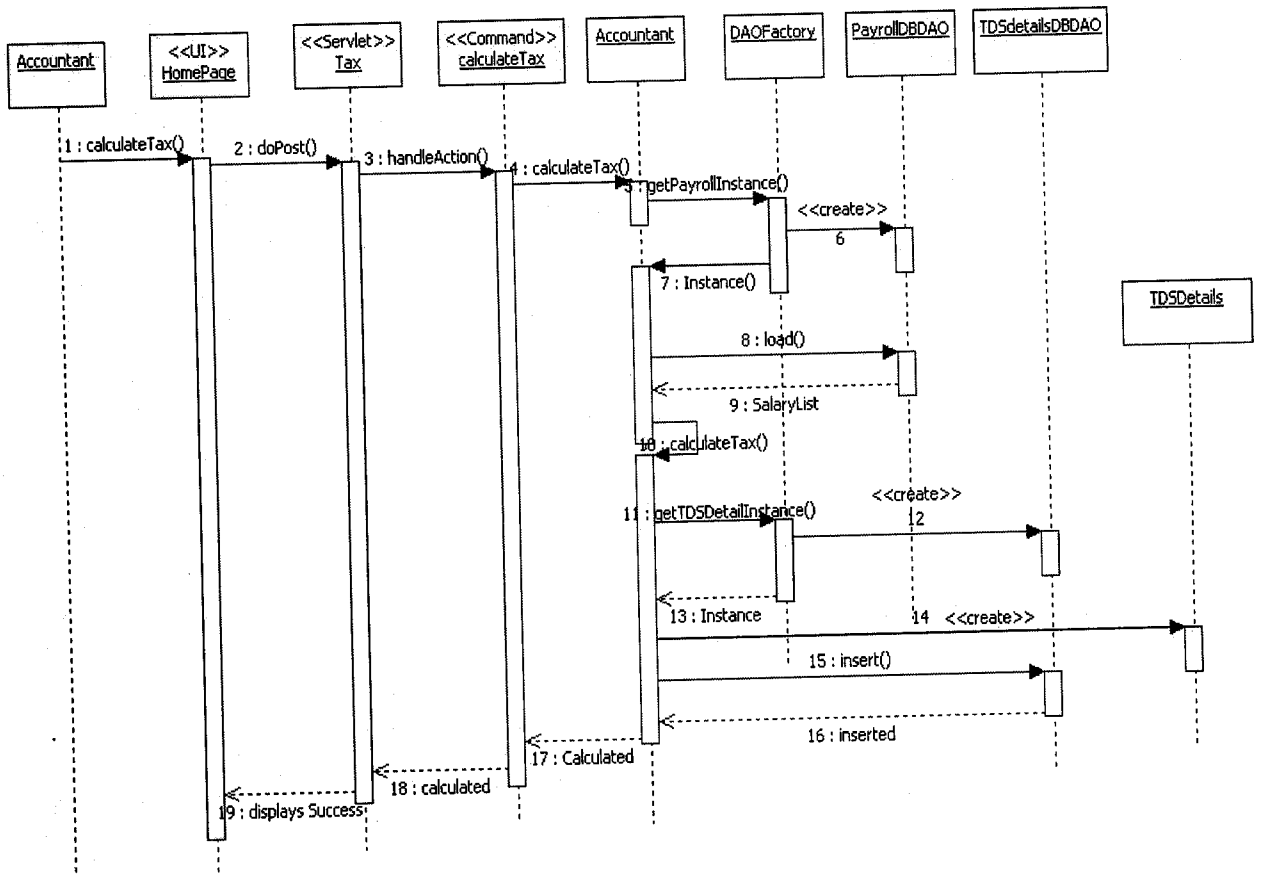


Figure 4.2.3.1 Sequence Diagram for Calculating Tax

4.2.4 COLLABORATION DIAGRAM

Collaboration diagram is similar to the sequence diagram. It shows the dynamic interaction of object in the system. The Figure 4.2.4.1 depicts the collaboration diagram for calculating tax.

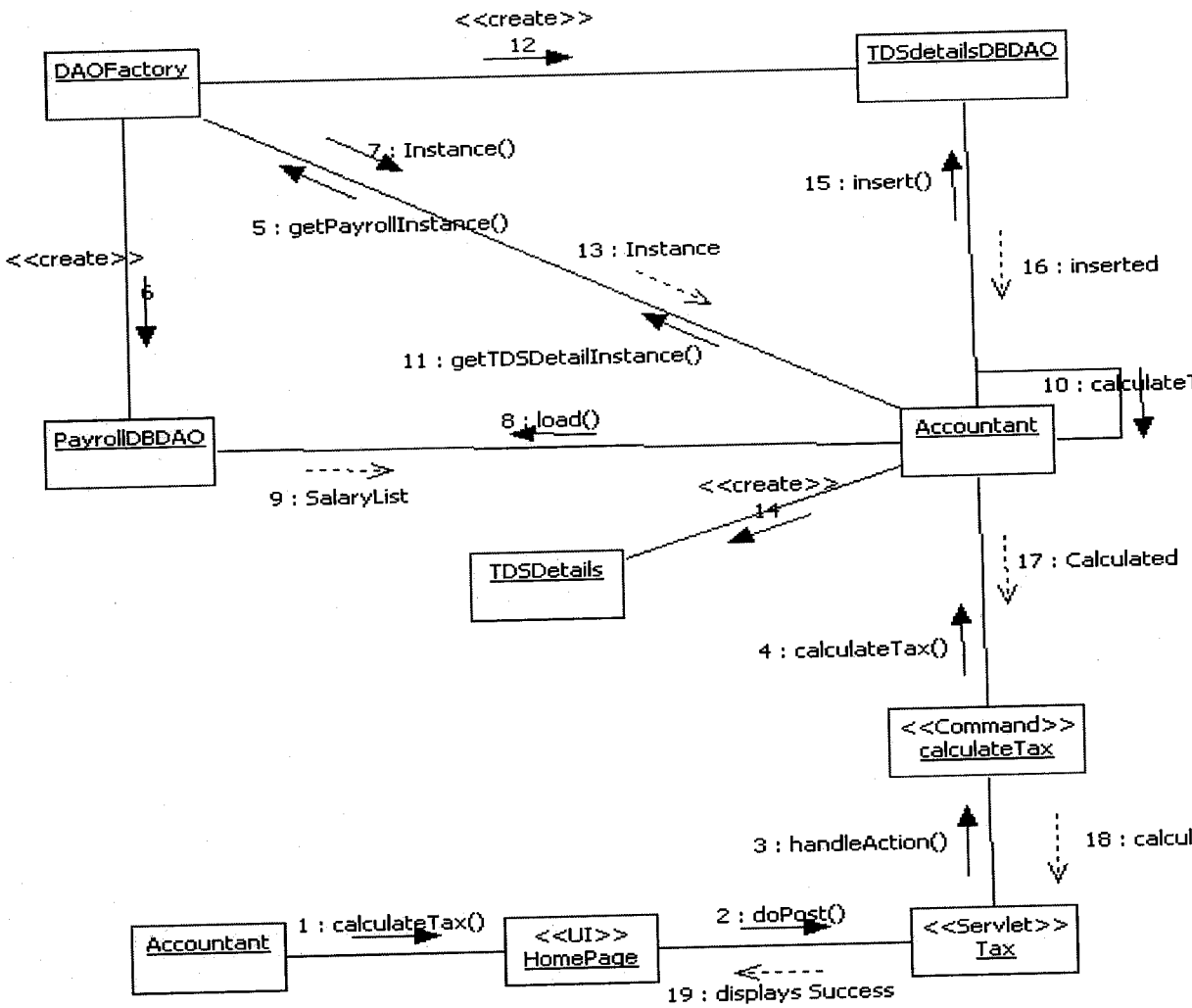


Figure 4.2.4.1 Collaboration diagram for Calculating Tax

4.2.4 CLASS DIAGRAM

The Figure 4.2.4.1 depicts the class diagram for the system.

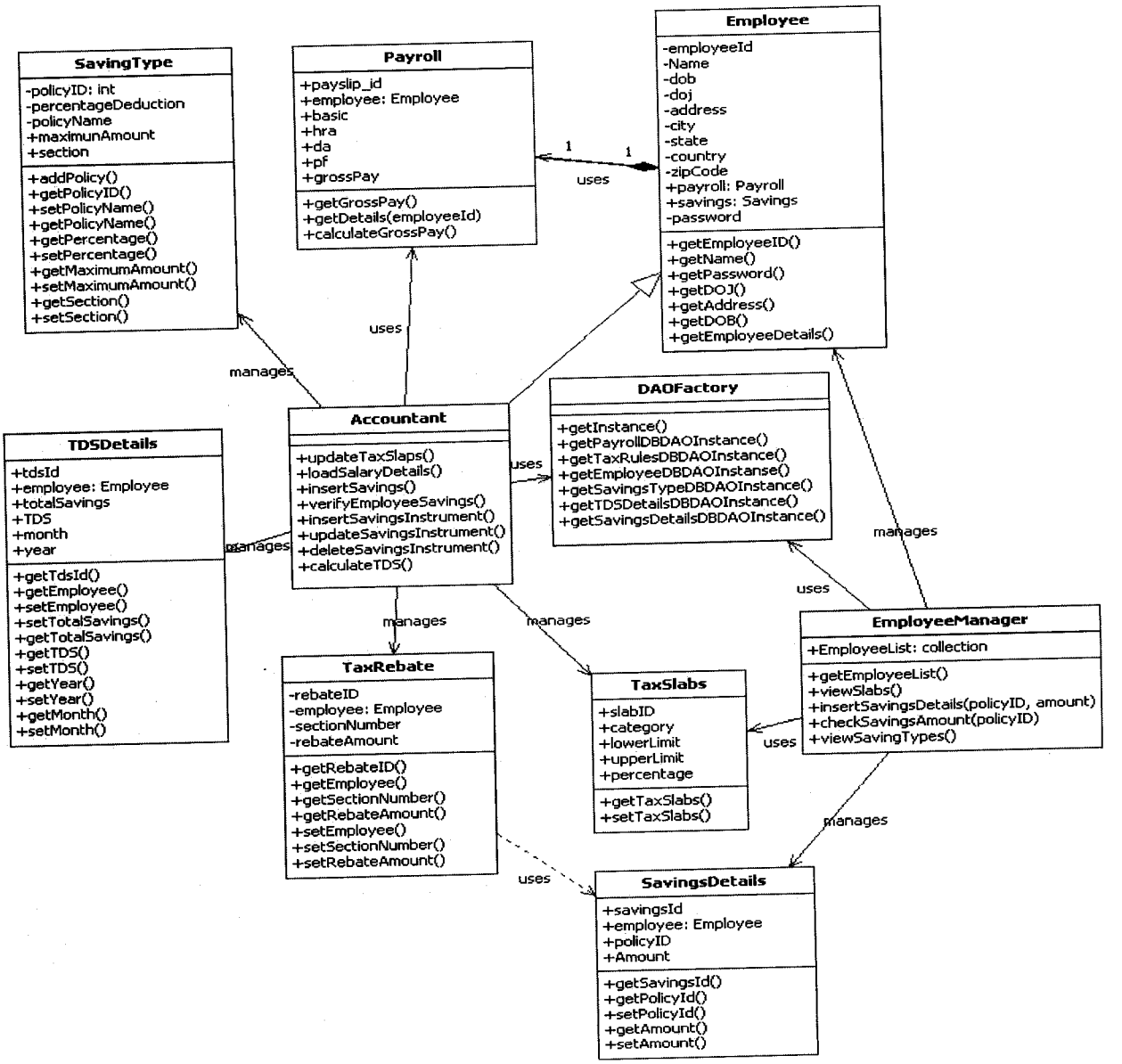


Figure 4.2.5.1 Class Diagram

4.3 DATABASE DESIGN

A database is a collection of inter-related data stored with minimum redundancy to serve many users quickly and efficiently. The general objective of database design is to make the data access easy, inexpensive and flexible to the user. The elegantly designed database plays a strong foundation for the whole system

The tables are normalized up to third normal form so that they can provide better response time, have data integrity, and secure.

4.3.1 TABLE STRUCTURE

The various tables used in the system are given from table 4.3.1.1 to 4.3.1.7

EMPLOYEE

This table is a Master Table, which contains all the employee details.

Fields	Type	Size	Constraints	Description
Employee_ID	Integer		Primary key	Employee Id of the Employee
First_Name	varchar	20		First name of the employee
Last_Name	varchar	20		Last name of the employee
Address	varchar	25		Address of the employee
City	varchar	20		City of the employee
State	varchar	20		State of the employee
Zip code	number	10		Zip code of the employee place
Country	varchar	20		Country of the employee
Password	varchar	20		Password for login
Phone Number	Integer	15		Phone number of the employee
DOB	Date			Date of Birth

Table 4.3.1.1

PAY_SCALE

This table is also a Master Table, which contains the monthly payroll details of the employees

Fields	Type	Size	Constraints	Description
Payslip_no	Integer		Primary Key	Pay slip number of the employee
Employee_ID	Integer		Foreign Key	Employee Identification number
Basic	Integer			Basic salary of the employee
HRA	Integer			House rent allowance
DA	Integer			Dearness allowance
PF	Integer			Provident Fund
OtherDeclaredIncome	Integer			Other Incomes
OtherTDS	Integer			TDS deducted by others

Table 4.3.1.2

TAX_SLABS

This table contains the current IT slabs.

Fields	Type	Size	Constraints	Description
Slab_ID	Integer	2	Primary Key	Id of the particular Category
Lower_Limit	Integer			The Lower Limit amount
Upper_limit	Integer			Upper Limit amount
Percentage	Integer	3		Percentage of tax for amount between the above range
category	char	1		Category of slab (i.e. Men or Women or Senior Citizen)

Table 4.3.1.3

TAX_REBATE

This table holds all the employees' rebate details

Fields	Type	Size	Constraints	Description
Rebate_ID	Integer		Primary Key	Identification number of the Rebate
Employee_Id	Integer		Foreign Key	Employee Identification Number
SectionNumber	Integer			Section Number
Amount	Integer			Rebate Amount

Table 4.3.1.4

SAVING_INSTRUMENT

This table contains the various instruments in which the employees can invest to save tax.

Fields	Type	Size	Constraints	Description
Policy_Id	Integer		Primary Key	Identification number of the policy
Policy	Varchar	20		Policy Name
Rebate_Amount	Integer			Maximum amount for the policy
Section	Integer			Section Number
Percentage_of_Deduction	Integer			Percentage of deduction allowed under this policy category.

Table 4.3.1.5

EMPLOYEE_SAVINGS

This table has the information about the employee's savings.

Fields	Type	Size	Constraints	Description
Savings_Id	Integer		Primary Key	Savings Id
Employee_Id	Integer		Foreign Key	Employee Id
Policy_id	Integer			Savings policy id
Amount	Integer			Amount of savings done under it
Status	string	10		Status of approval (updated by accountant)

Table 4.3.1.6

EMPLOYEE_TDS

This table is used to store the TDS details of the employees

Fields	Type	Size	Constraints	Description
TDS_ID	Integer		Primary key	Identification Number of the TDS details
Employee_Id	Integer		Foreign Key	Employee id
TDS	Integer			TDS for monthly deduction
Month	Integer			Current Month
Year	Integer			Current financial year

Table 4.3.1.7

CHAPTER 5

SYSTEM TESTING AND IMPLEMENTATION

The system testing is the stage of implementation, which ensures that the system works accurately and efficiently before live operation commences. Testing plays a major part in software development. Once the customer has specified the requirements the system has to be developed so as to satisfy those requirements and this is ensured by means of testing.

In the Income Tax Generator various testing techniques are used to test the modules individually and also the system as a whole to determine the quality of the system. Some of the modules are as follows.

5.1 TESTING OBJECTIVES

The testing objectives are,

- Testing is a process of executing a program with the intent of finding an error.
- A successful test is one that uncovers an as yet undiscovered error.

Testing demonstrates that the software functions work according to specifications. In addition data collected from testing provides a good indication of software reliability and some indication of software quality as a whole.

5.2 TEST ENVIRONMENT

Testing will be done using a single web site and a minimum of two clients. The web site will be set up on a Windows machine using Tomcat Web server. Clients will also be Windows

machines with a web browser. Testing will be done on Internet Explorer and Mozilla Firefox browsers.

5.3 TESTING METHODS

The application functionality will be tested using manual methods. Testing Methods applied in Income tax Generator are,

1. Unit Testing
2. Integrated Testing
3. Validation Testing
4. User-Interface testing
5. Stress Testing

5.3.1 UNIT TESTING

Unit testing focus on the verification effort of the smallest unit of software design such as the software component or the module. The modules of the system are tested separately. This testing is done at the programming stage itself.

In the Income tax generator system there are six modules and each of them are tested individually to determine errors. The boundary conditions of the modules are tested to ensure that the module operates properly.

5.3.2 INTEGRATION TESTING

Integration Testing is a process of integrating the various modules of the system to form a complete system. It is a systematic technique for constructing the software architecture while at the same time it uncovers the errors associated with interfacing.

In Income Tax Generator system bottom-up integration testing is done, in which the modules are combined one by one and tested and finally tested as a whole. The interdependencies between the modules are tested and the bugs are traced and corrected.

5.3.3 VALIDATION TESTING

The requirements established as a part of software requirements analysis are validated against the software that has been constructed, this is validation testing. It provides final assurance that software meets all functional, behavioral and performance requirements. Black box testing techniques are used exclusively during validation.

In the Income Tax Generator system various test cases are generated to test the whether the system meets the given requirement. the test cases are shown from Table 5.4.1.1

5.3.4 STRESS TESTING

Stress testing is executing the application in such a way it demands abnormal quantity of resources. The amount of data load is increased and tested for results.

In the Income Tax Generator system, the software is tested with various test cases. The system is tested by asking 50 users to log in to the system and asked to access the same data. By doing so the stress control for peak hours and peak loads are tested.

5.3.5 USER INTERFACE TESTING

User interface testing is used to test the GUI whether it is user-friendly or not. The Income Tax Generator System GUI is tested with various test cases. It is tested whether user friendly and understandable messages are thrown when an error occurs. The system is also tested whether is redirects the user to the appropriated pages when an operation is done or failed.

5.4 TEST CASES

5.4.1 FEATURES TO BE TESTED

The following functional features will be tested:

- Login
- Updating slabs
- Inserting savings detail
- Verify and update status
- Insert other savings details
- Calculate tax
- Logout

Additionally testing will be done simultaneously from multiple clients to ensure that all data changes are done online

Use case	Test case	Expected result
Log in	Log in as regular user with correct name and password	The main page with several facilities should be available to the user. Facility to add or update the TDS details and Salary details should not be available
	Log in as regular user with incorrect name and password	An error message is displayed and no facilities are available
	Log in as admin user with correct name and password	view , add savings details and update savings details , calculate TDS facilities are available
	Log in as admin user with incorrect name and password	An error message is displayed and no facilities are available

Enter savings details Precondition: Employee user logged in	Add a new savings detail with corresponding savings instruments all set and other details correct	A new record is added and a message is displayed showing inserted.
	Add a new savings detail without and the savings instrument set / or other details incorrect	An error message is displayed and no information is added
	Add a new saving details with corresponding savings instrument set and other details correct Add the same record again from the same client	The first record is added. The second record is not added, an error message is displayed.
Verify and update Status Precondition: admin user logged in	Select the employee savings record and set the status. Save the details.	A message is displayed showing updated.

Update savings instruments Precondition: admin user logged in	Add all the details of the new savings instrument.	The new savings instrument is added and the list of policies is listed out.
	Add the same policy once again	Message is displayed that the policy already exists
Insert/Update slabs Precondition: admin user logged in	Add new slab with lower limit , upper limit, percentage and category	The new slab is added and displayed.
	Add new slab with upper limit lesser than the lower limit.	An error message is displayed.
Delete Slabs	Select the slab id from the drop down list and click delete	The system will delete the particular record and will display all the slabs

Update other income detail Precondition: Employee user logged in	Add other income details and save the details	The new record is added and a message id displayed as 'updated' and the details are displayed.
Calculate Tax Precondition: Accountant user logged in	Select the 'Calculate Tax' option	The system will display a report of all the employee with their TDS for the Current month
	Select the 'Calculate Tax' option when already calculated	The system will show its already calculated.
Logout	Invoke the logout mechanism	Facilities are no longer available
	Logout, and try to use any facility using browser back button	Error message should be displayed along with login screen

Table 5.4.1.1

5.5 IMPLEMENTATION

System Implementation is the stage of the project when the theoretical design is turned into a working system. At this stage the main workload, the greatest upheaval and the major impact on existing practices shifts to the user department. If implementation stage is not carefully planned and controlled, it can cause chaos. Thus it can be considered to be the most crucial stage in achieving a successful new system and in giving the user confidence that the new system will work and be effective.

The new system was uploaded in the company's site. The system was developed as a supporting system only, so it was easy to implement.

5.5.1 USER TRAINING

In this phase, user training is critical for minimizing reluctance to change and giving new system a chance to prove its worth. The new system may be replacing the existing system completely or partially. In either case it is essential to provide a reliable system to meet organizational requirements.

User training has been given to the users of the system. The training was conducted on how to use the system. A special training was given to the accountant on how to handle errors and requests. The user manual is prepared and also given to the users for better understanding of the system.

CHAPTER 6

FUTURE ENHANCEMENT

The system has so many advanced features than the existing system by means of its user friendliness. Apart from the features given in the developed system, further extension and redesign can be done according to the client requirements. But it can draw few possible extension modules as a part of initial level suggestions to give a general idea to the clients who are implementing the project and those suggestions can be consolidated as follows.

Extensions can be used to fill in the Form-16 Automatically by the system, where now only the format of Form-16 and related forms are available. The employee has to print the form and fill manually.

APPENDIX A

SCREEN LAYOUTS

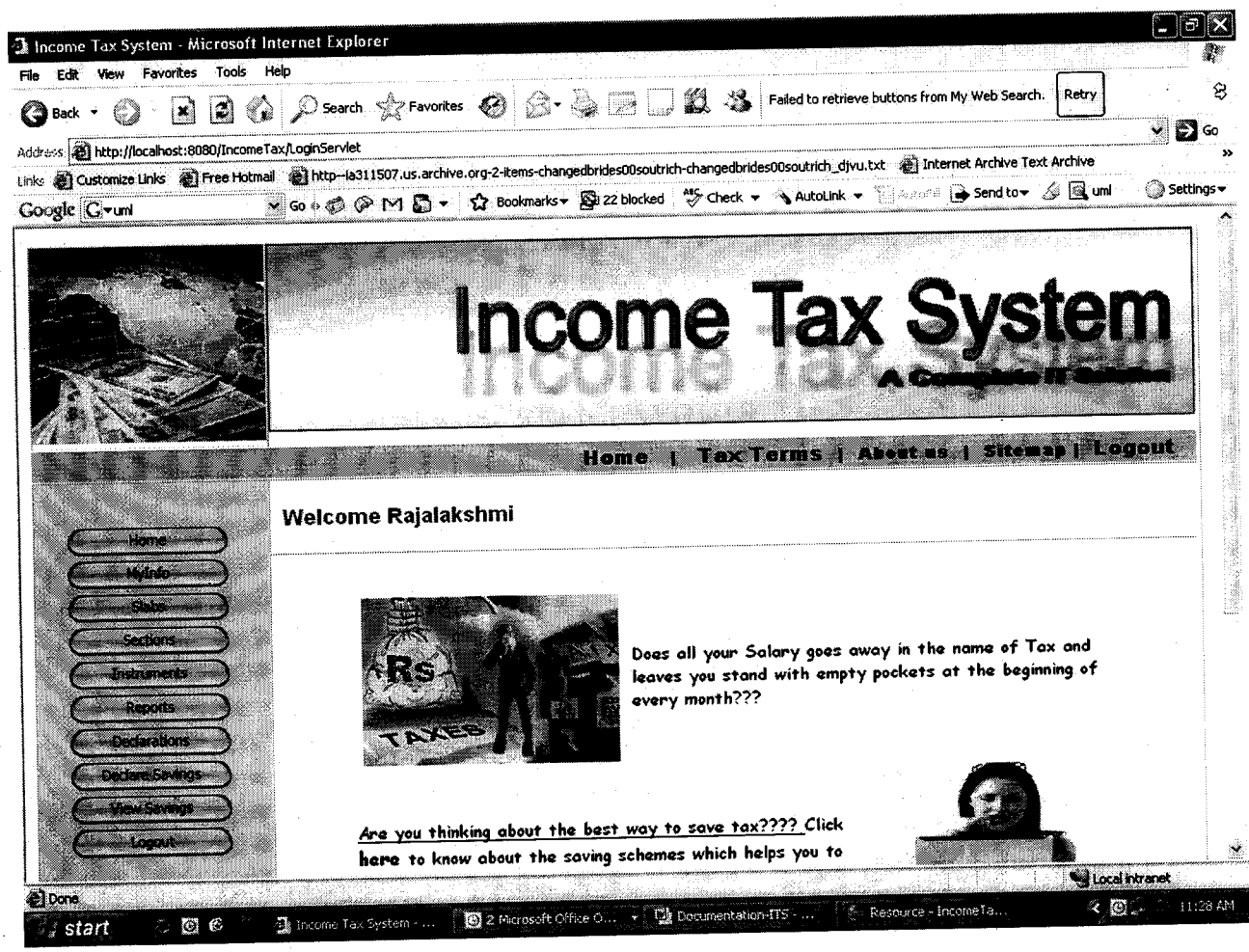


Figure A.1 Home Page

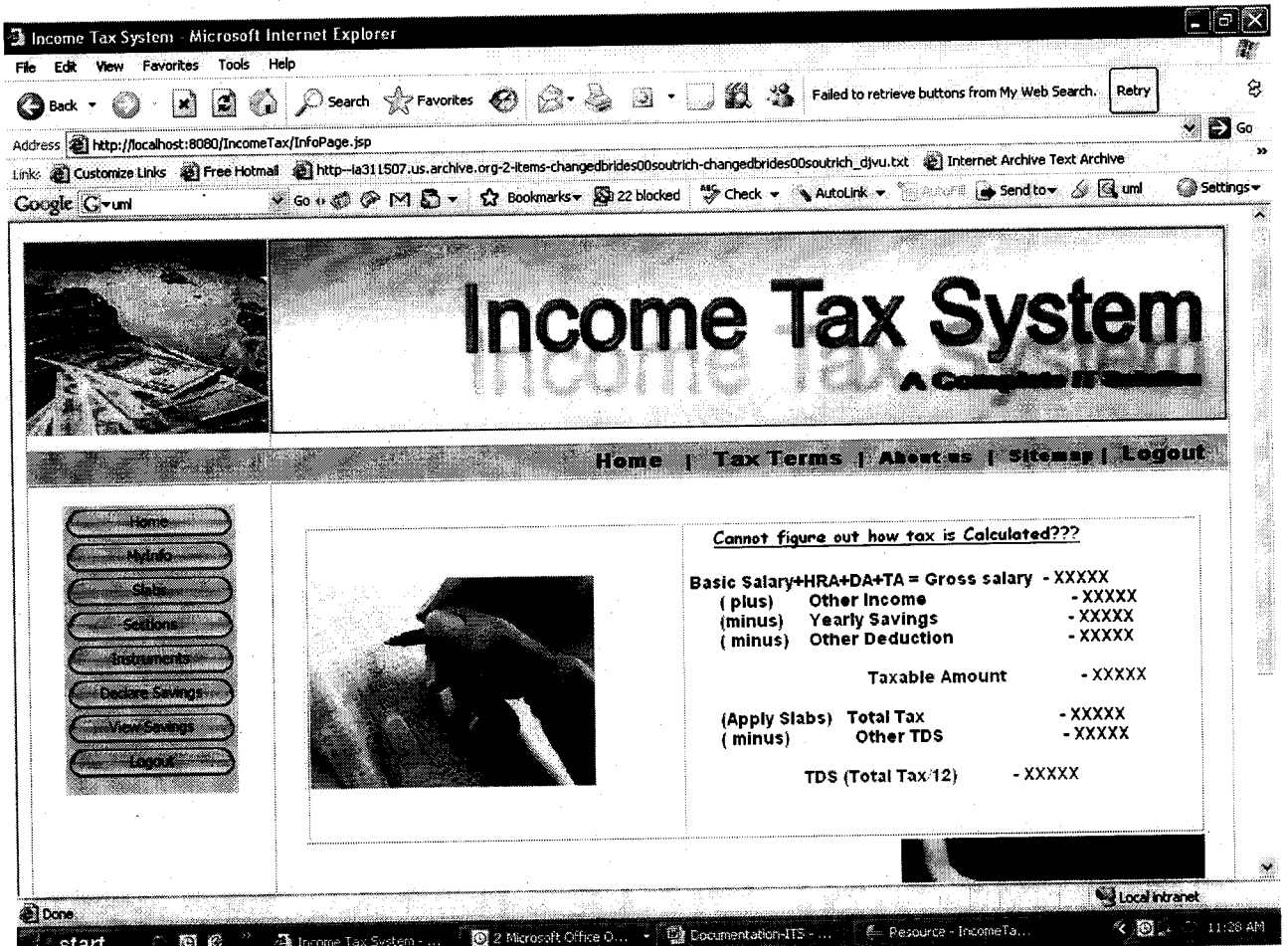


Figure A.2 Information Page

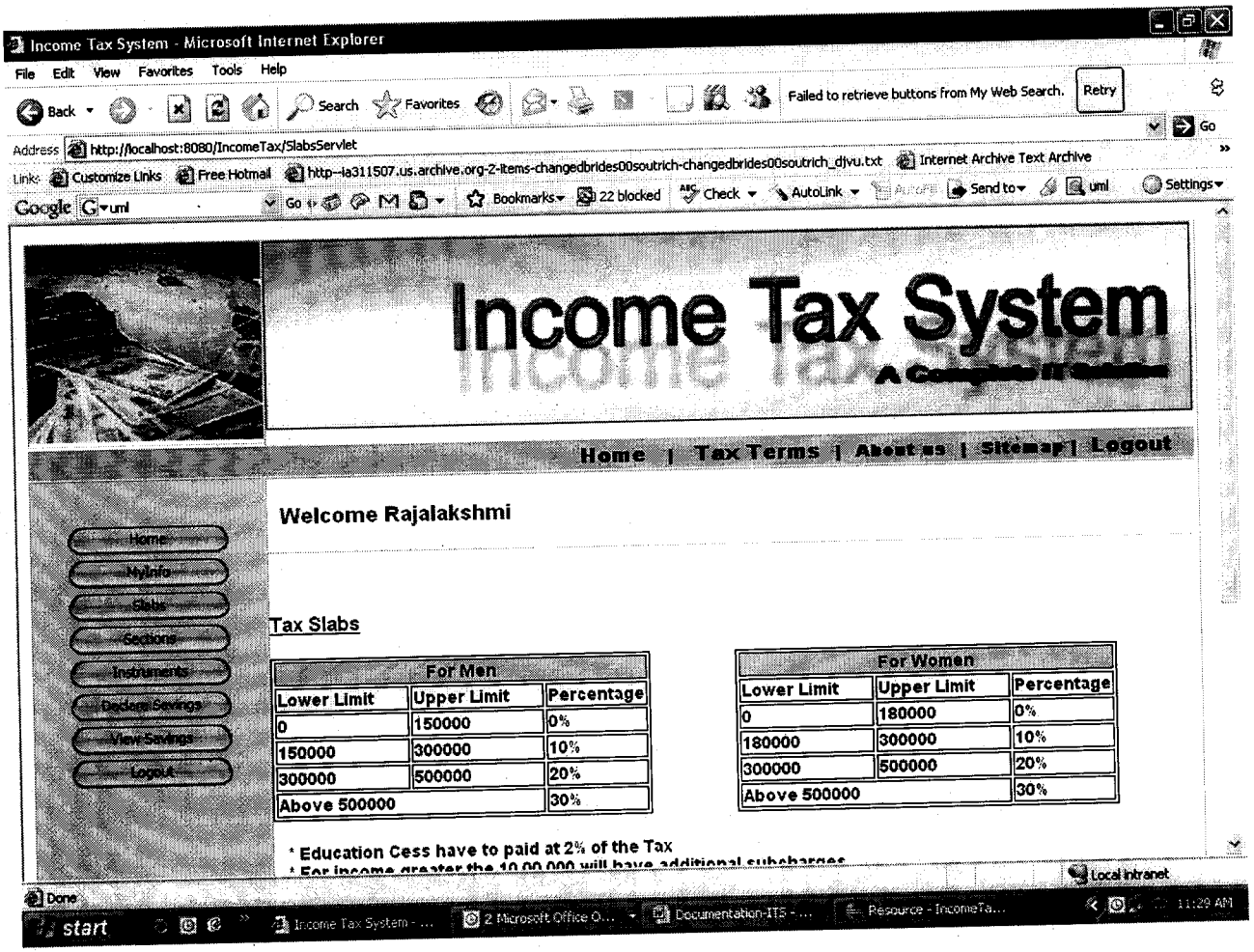


Figure A.3 Tax Slabs Page

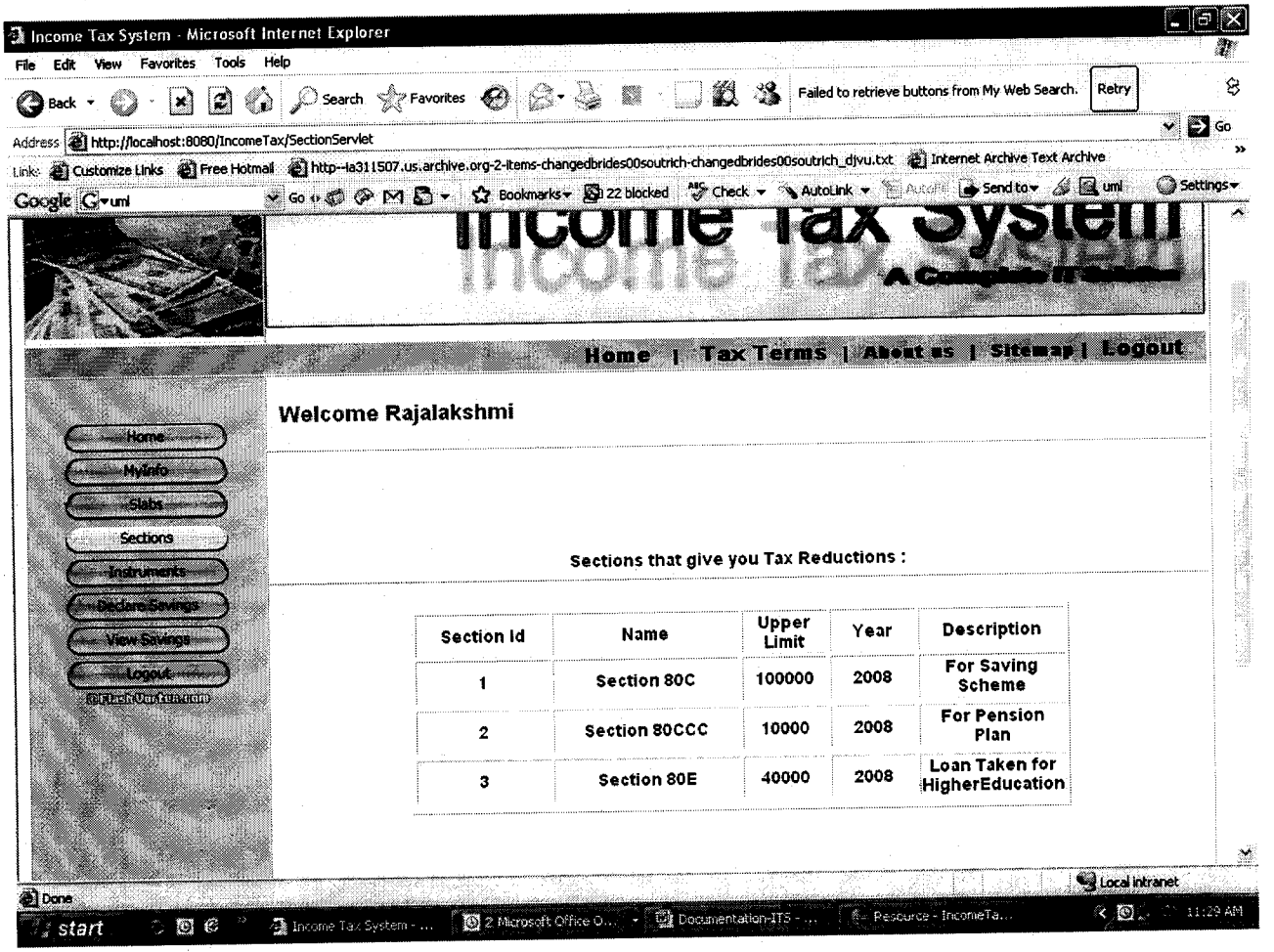


Figure A.4 Sections Page

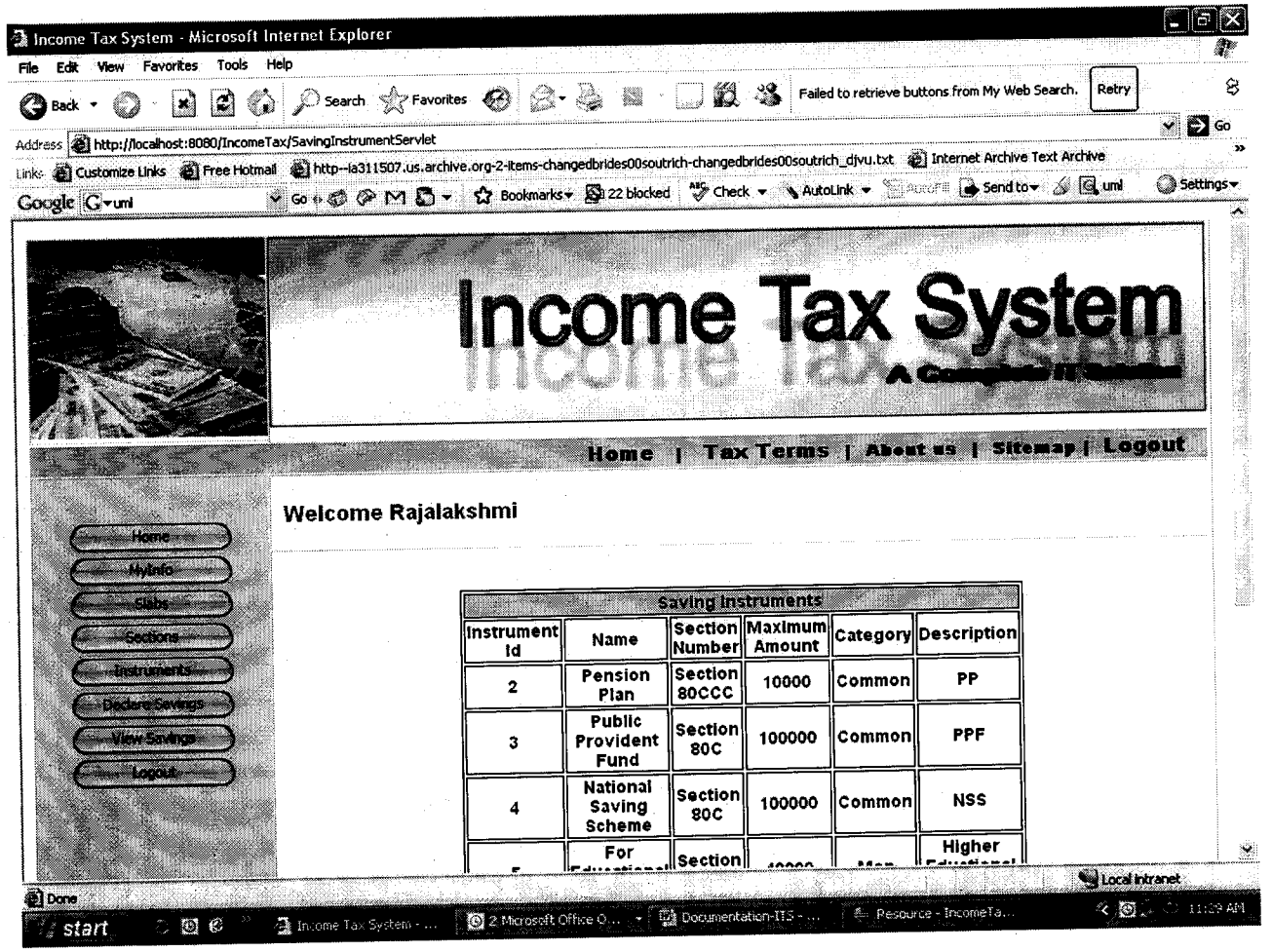


Figure A.5 Saving Instruments Page

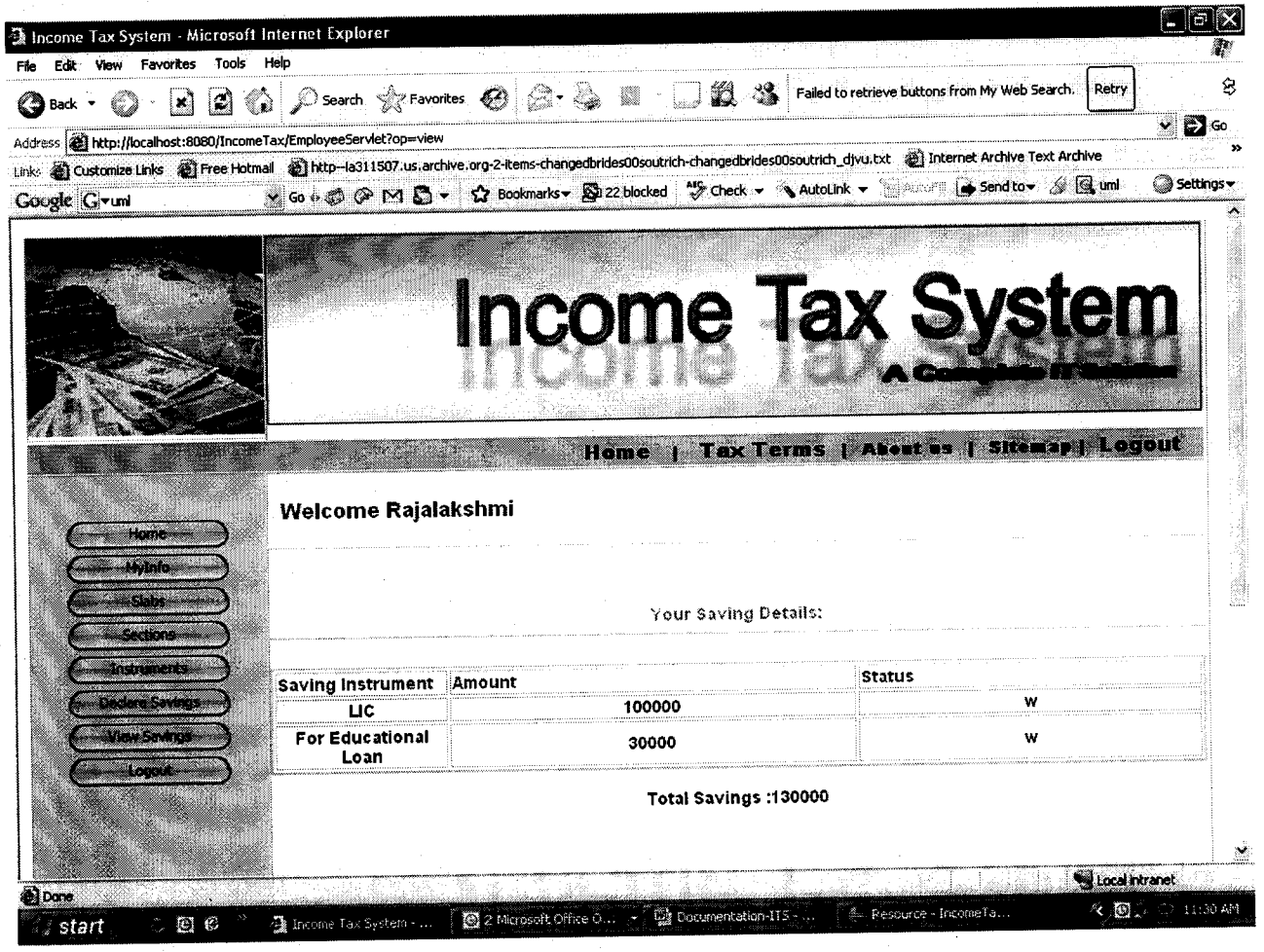


Figure A.6 Savings details Page of the employee

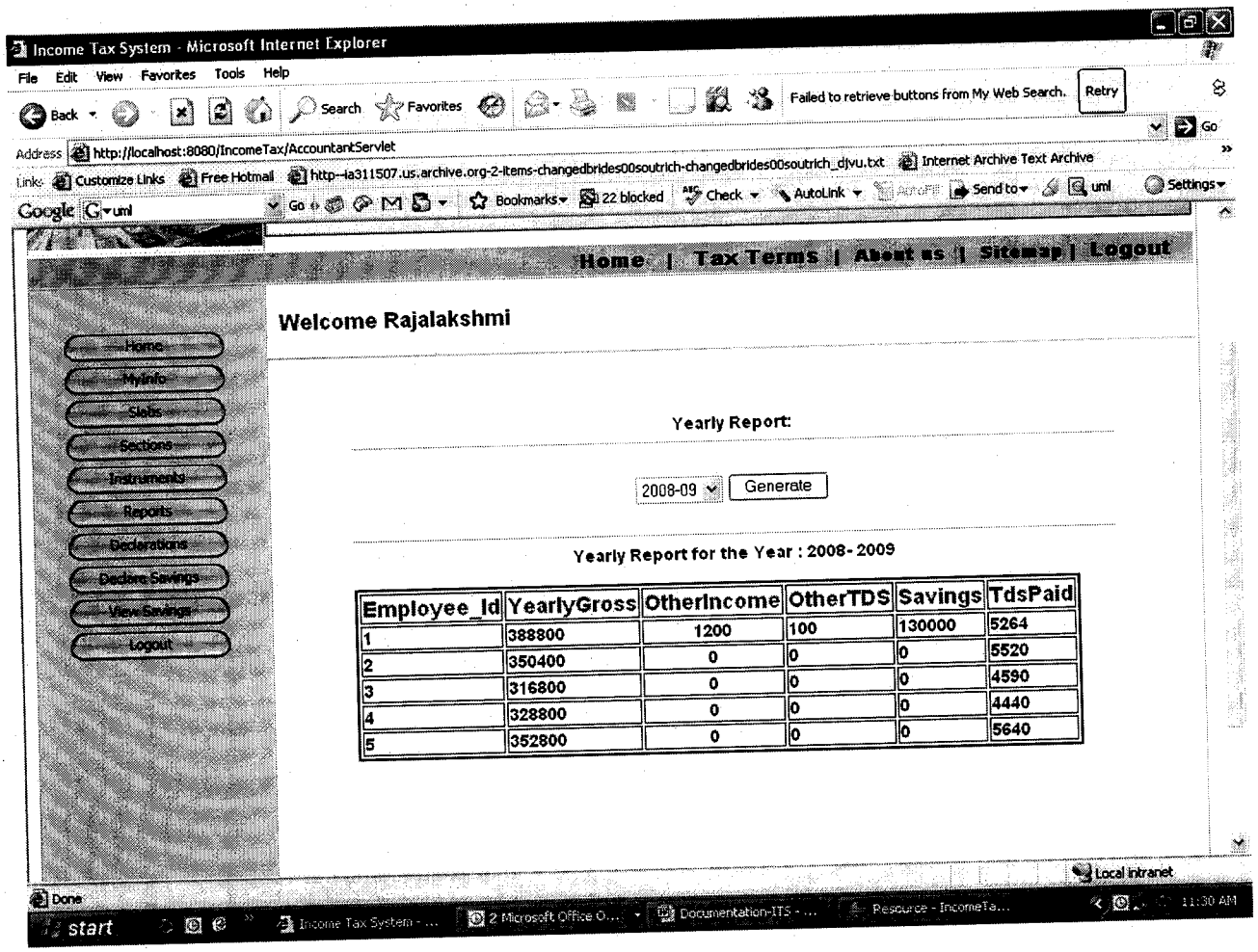


Figure A.6 Yearly report Page of all employees

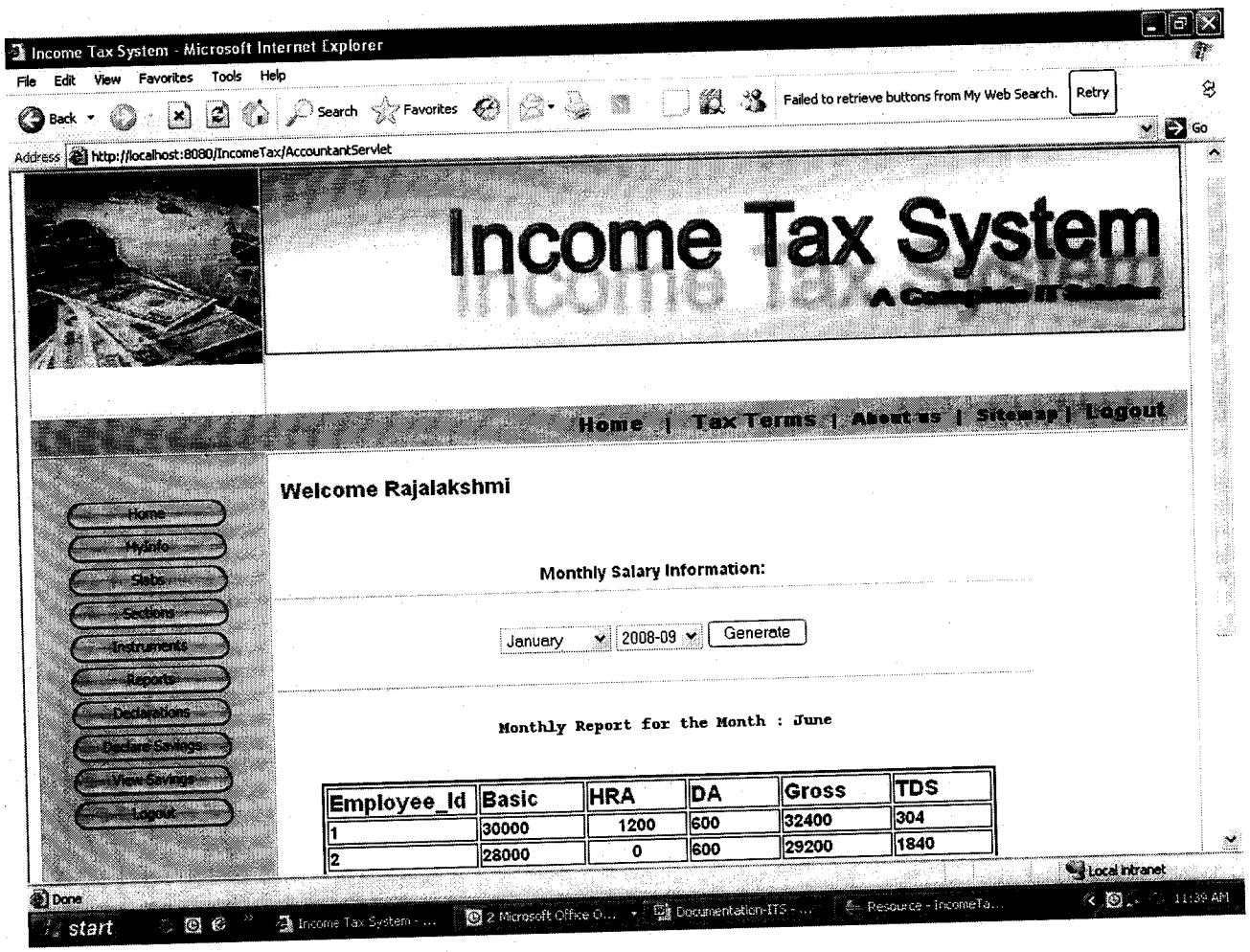


Figure A.7 Monthly Salary Report for the selected month

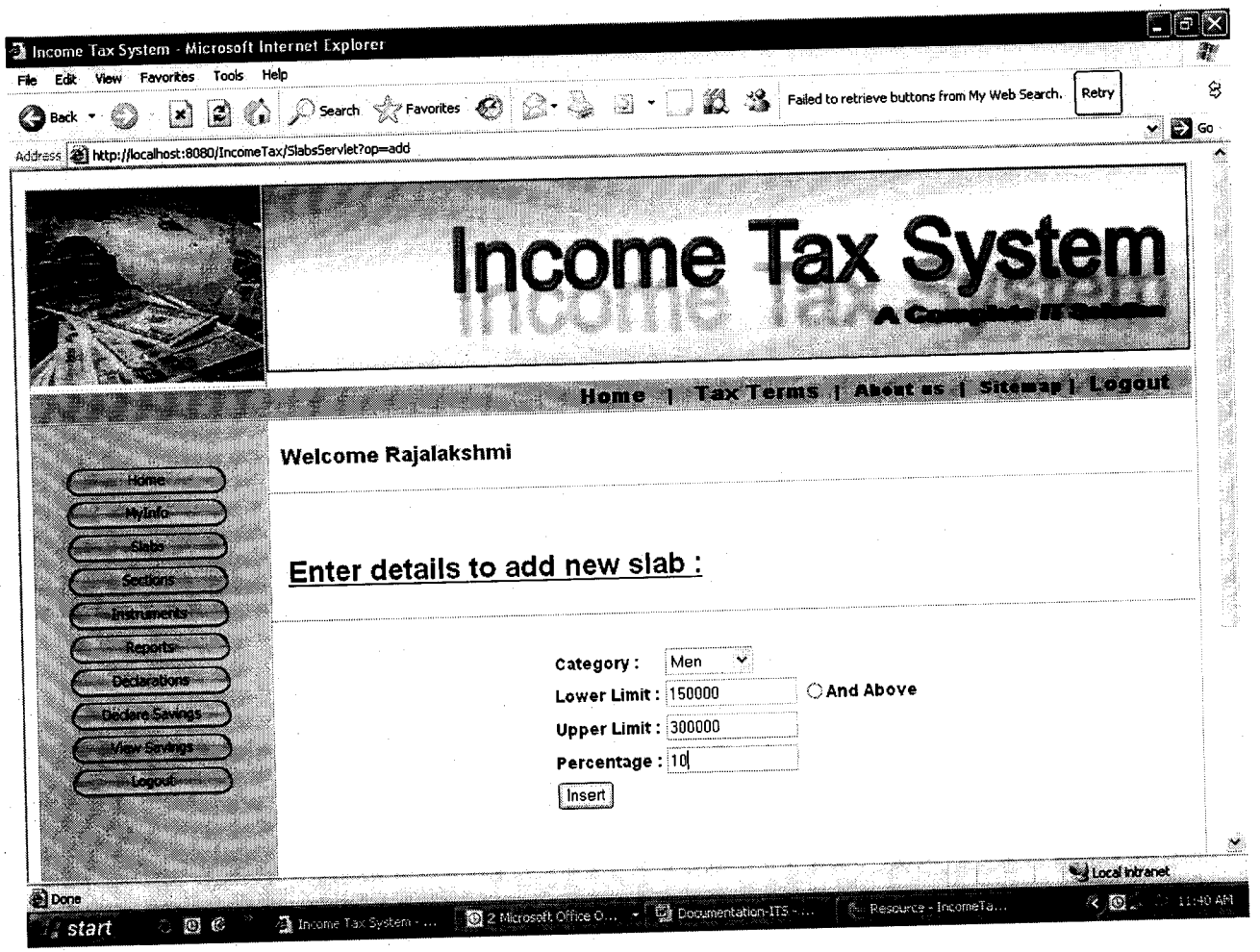


Figure A.8 Adding new Slabs

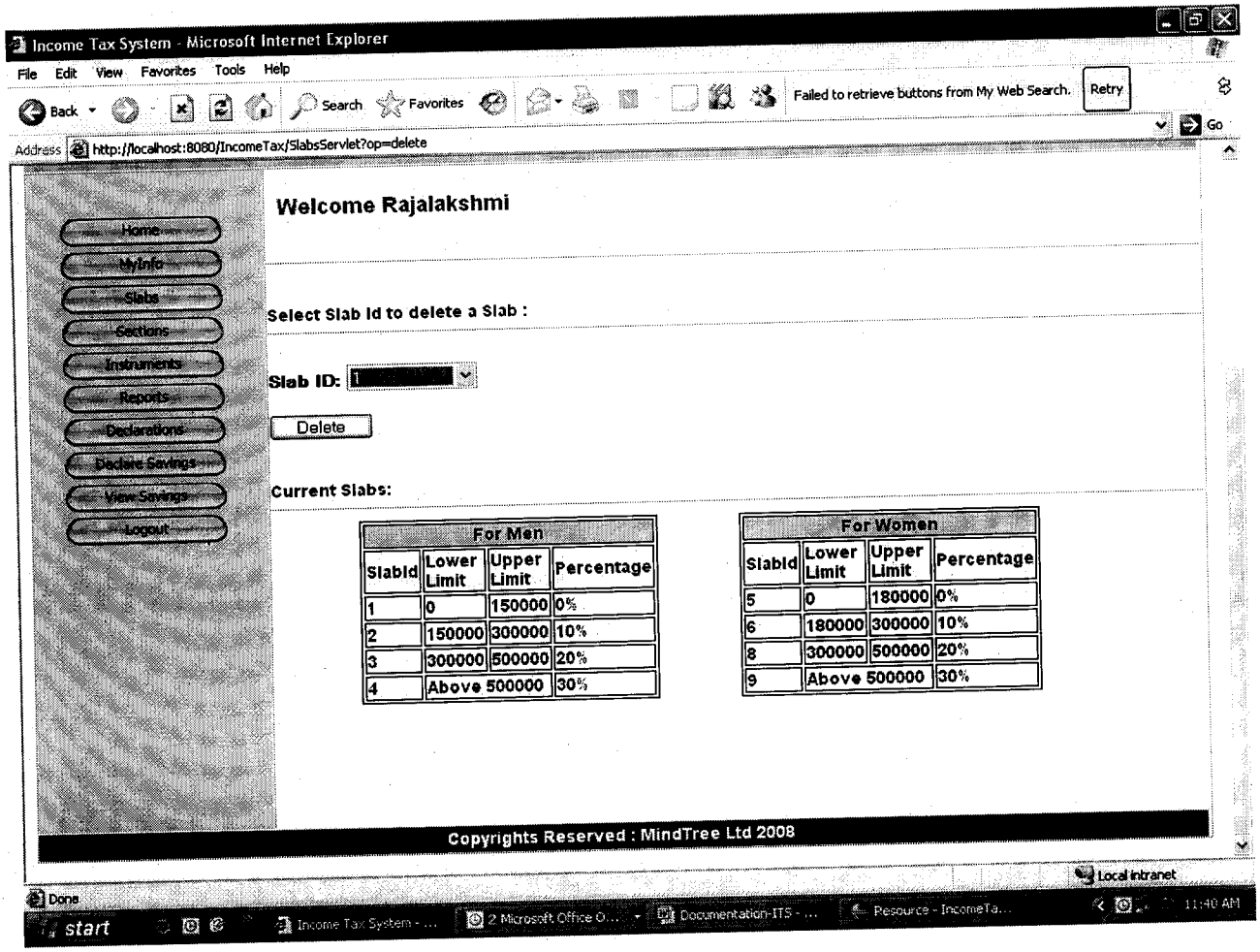


Figure A.9 Deleting a Slab

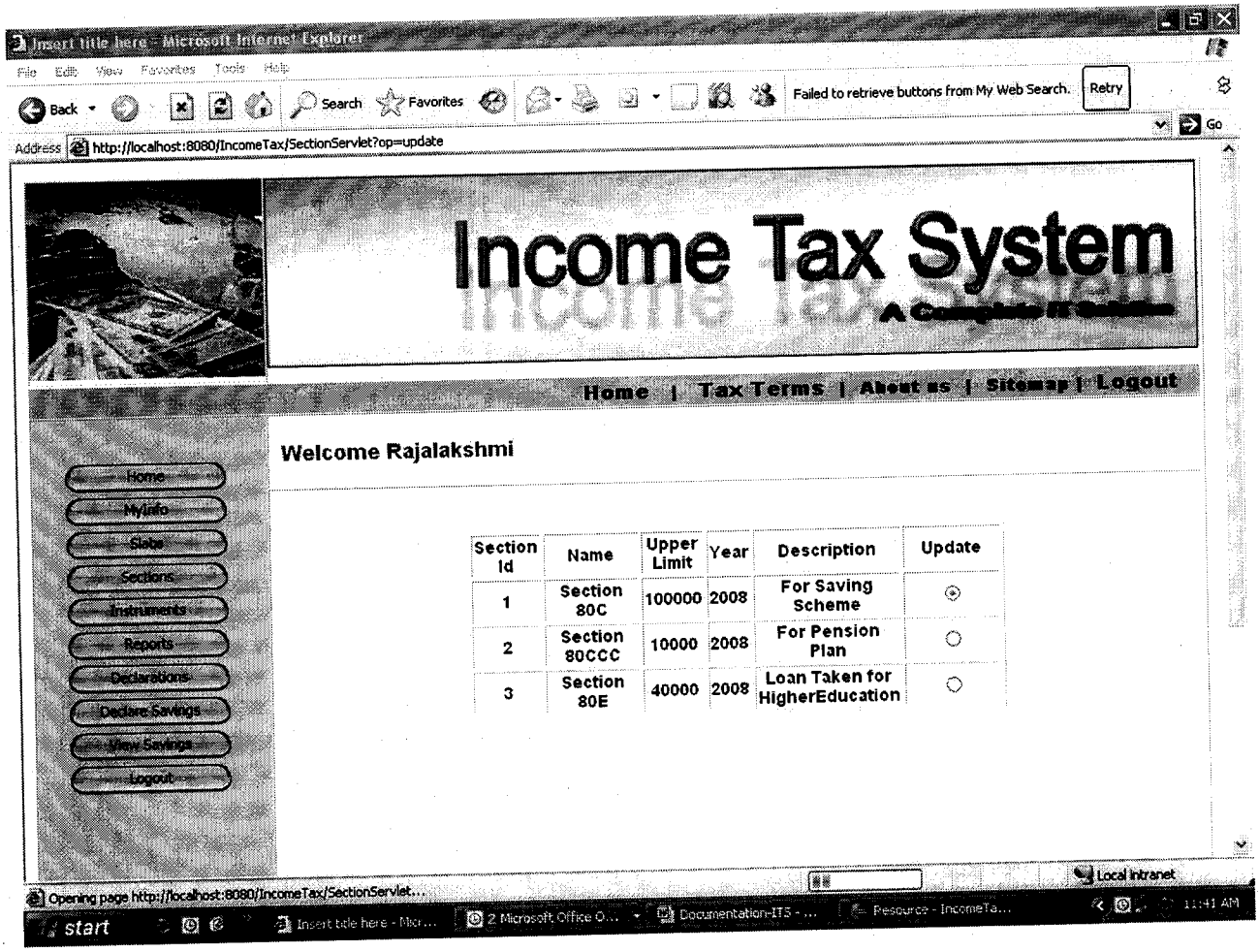


Figure A.10 Updating a Section (a)

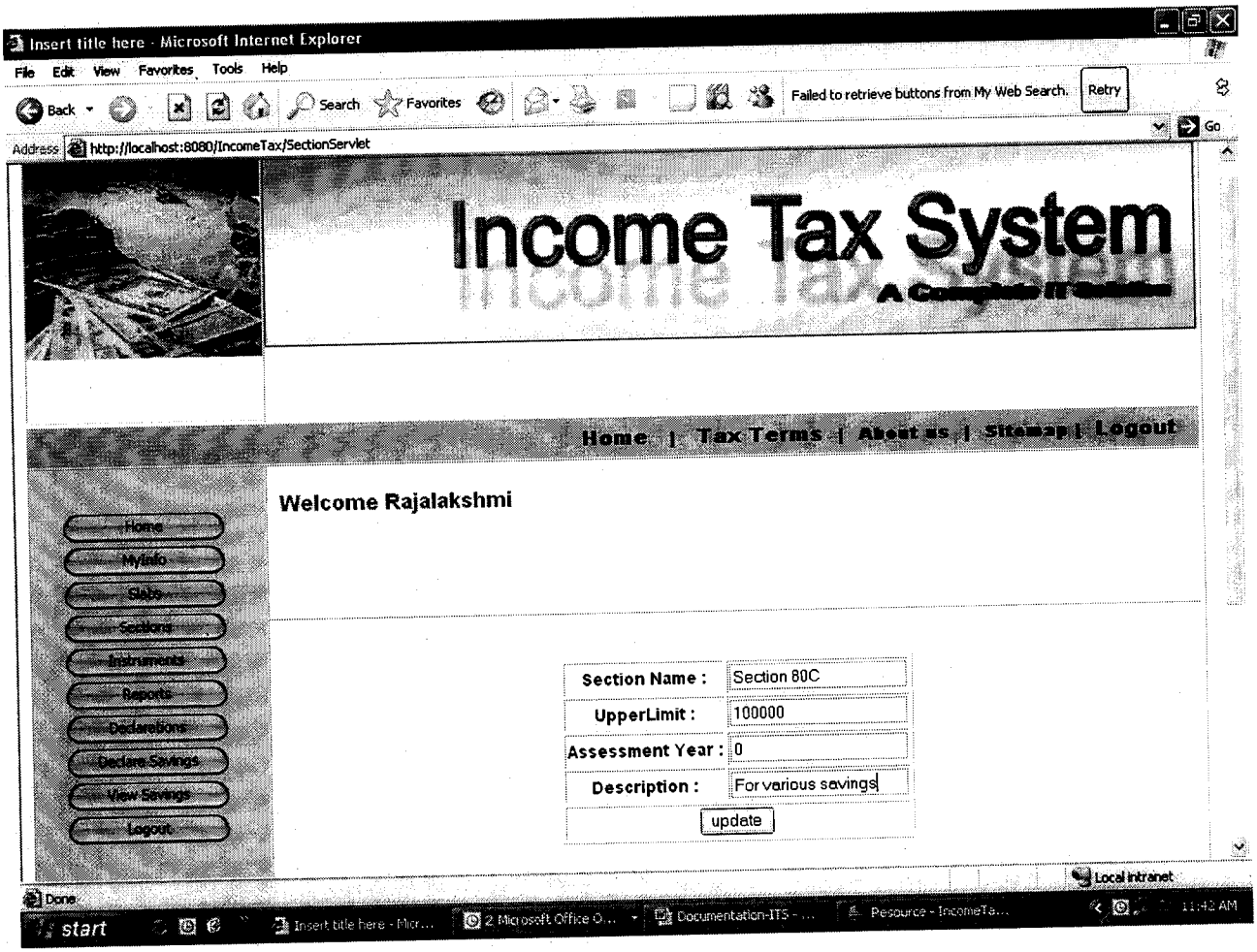


Figure A.11 Updating section (b)

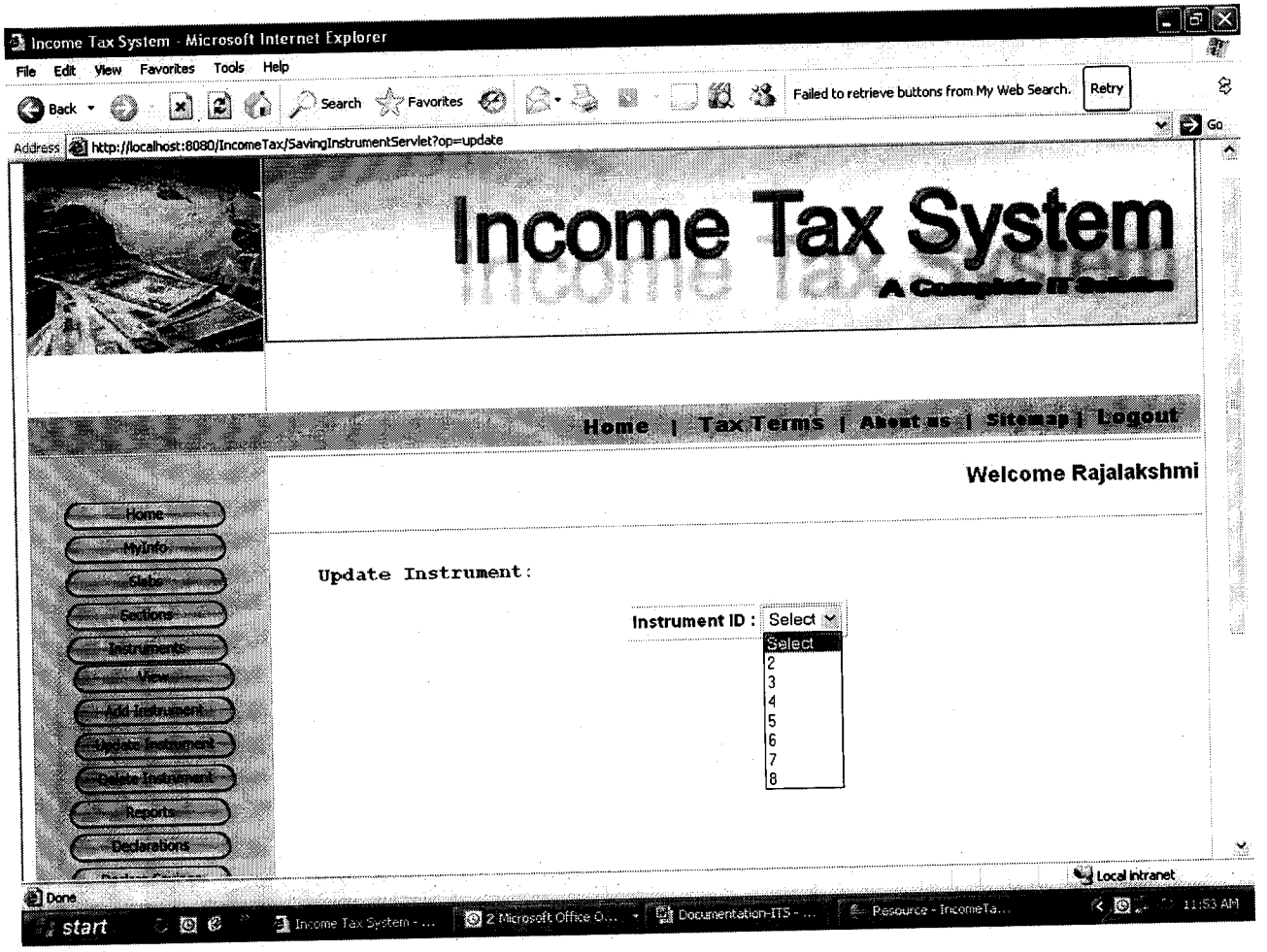


Figure A.12 Updating Instrument (a)

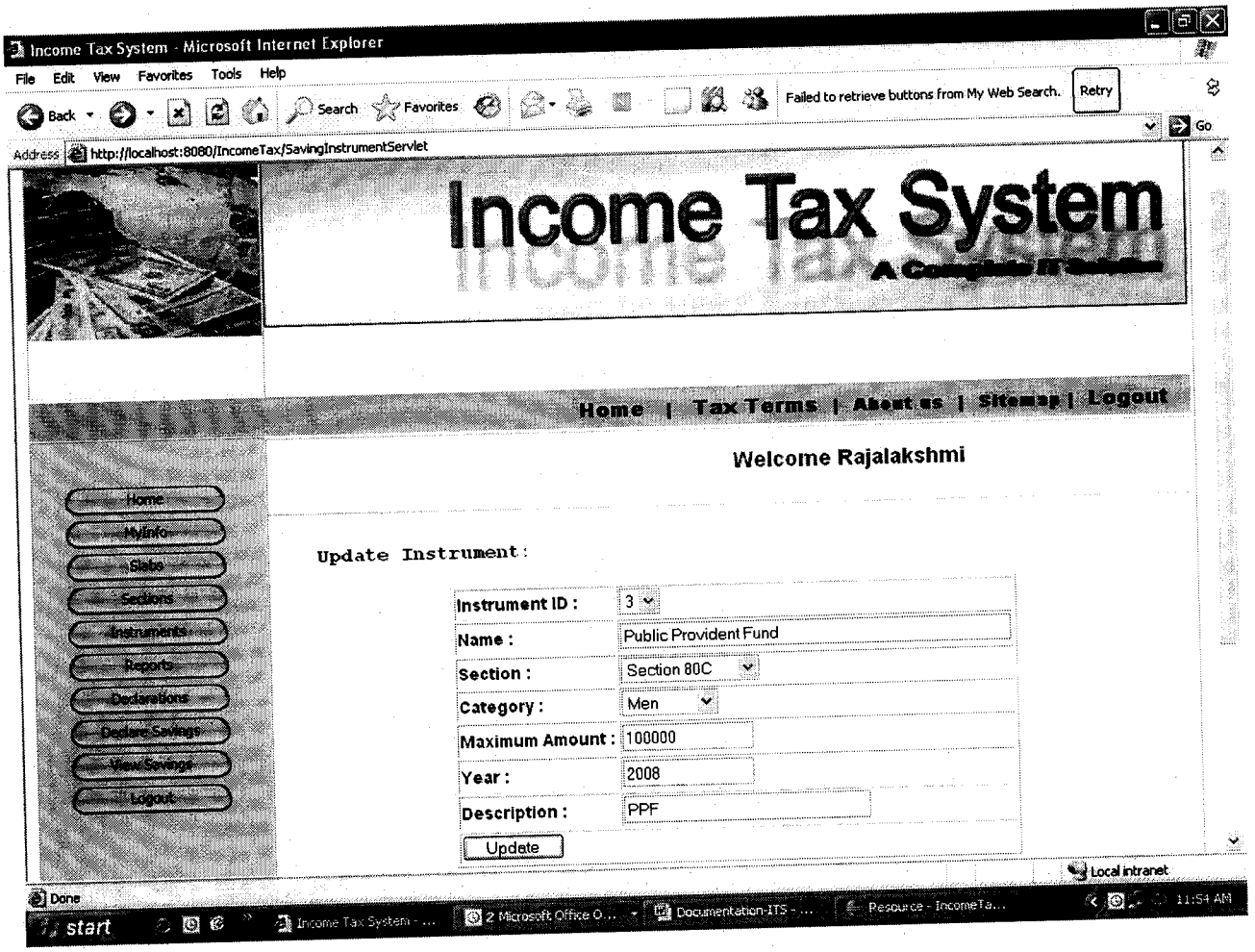


Figure A.13 Updating Instrument (b)

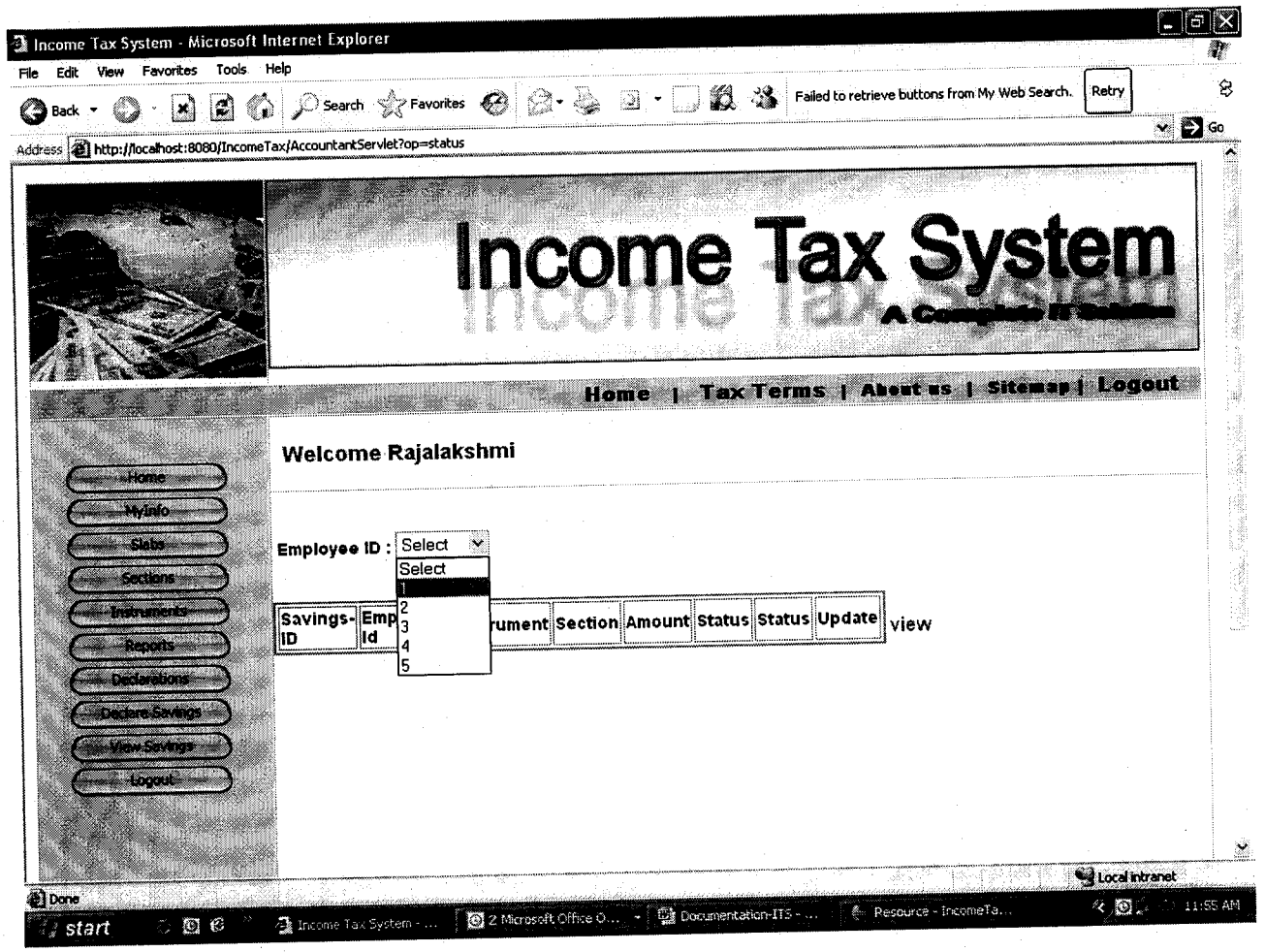


Figure A.14 Updating status of employee saving after verifying proofs (a)

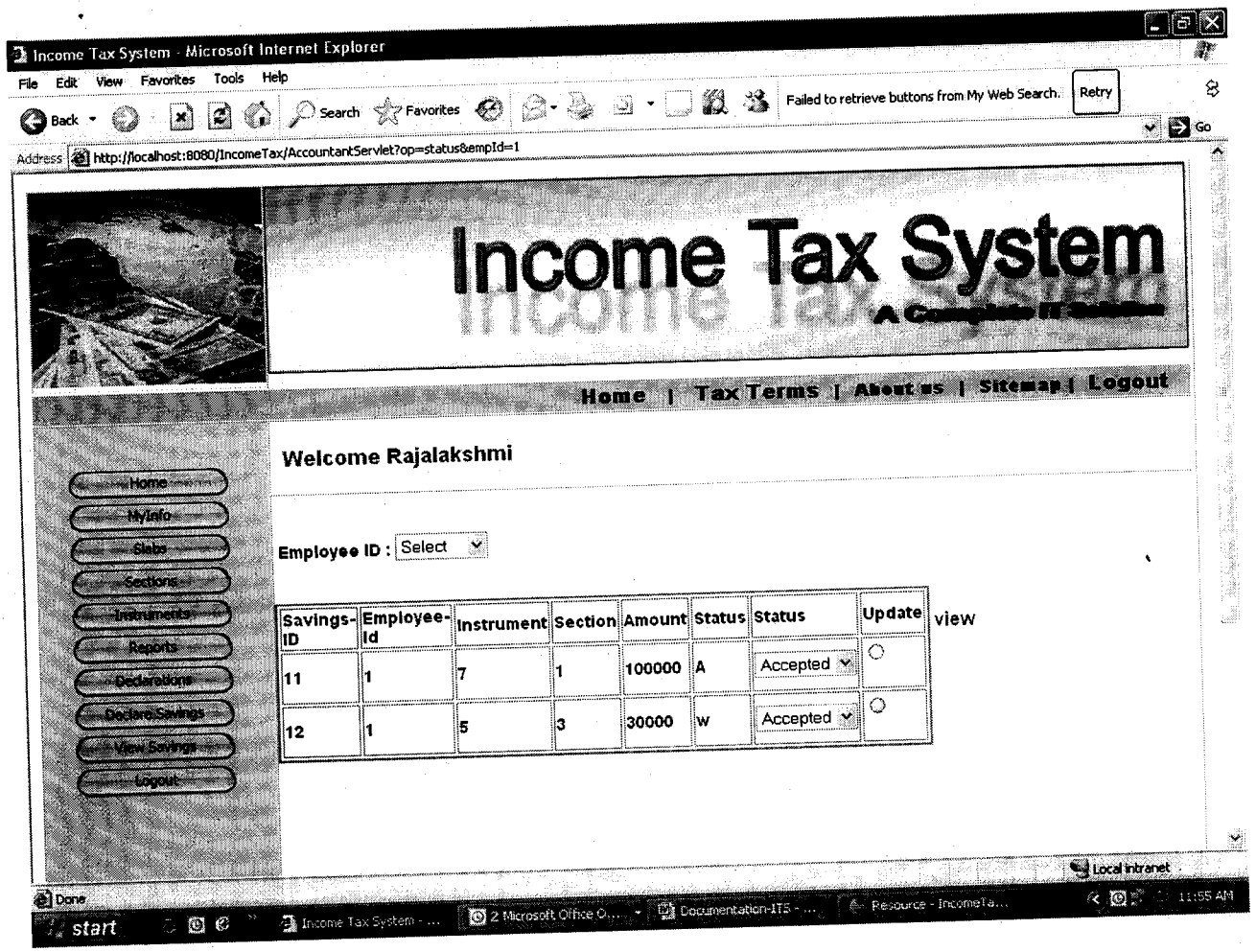


Figure A.15 Updating status of employee saving after verifying proofs (b)

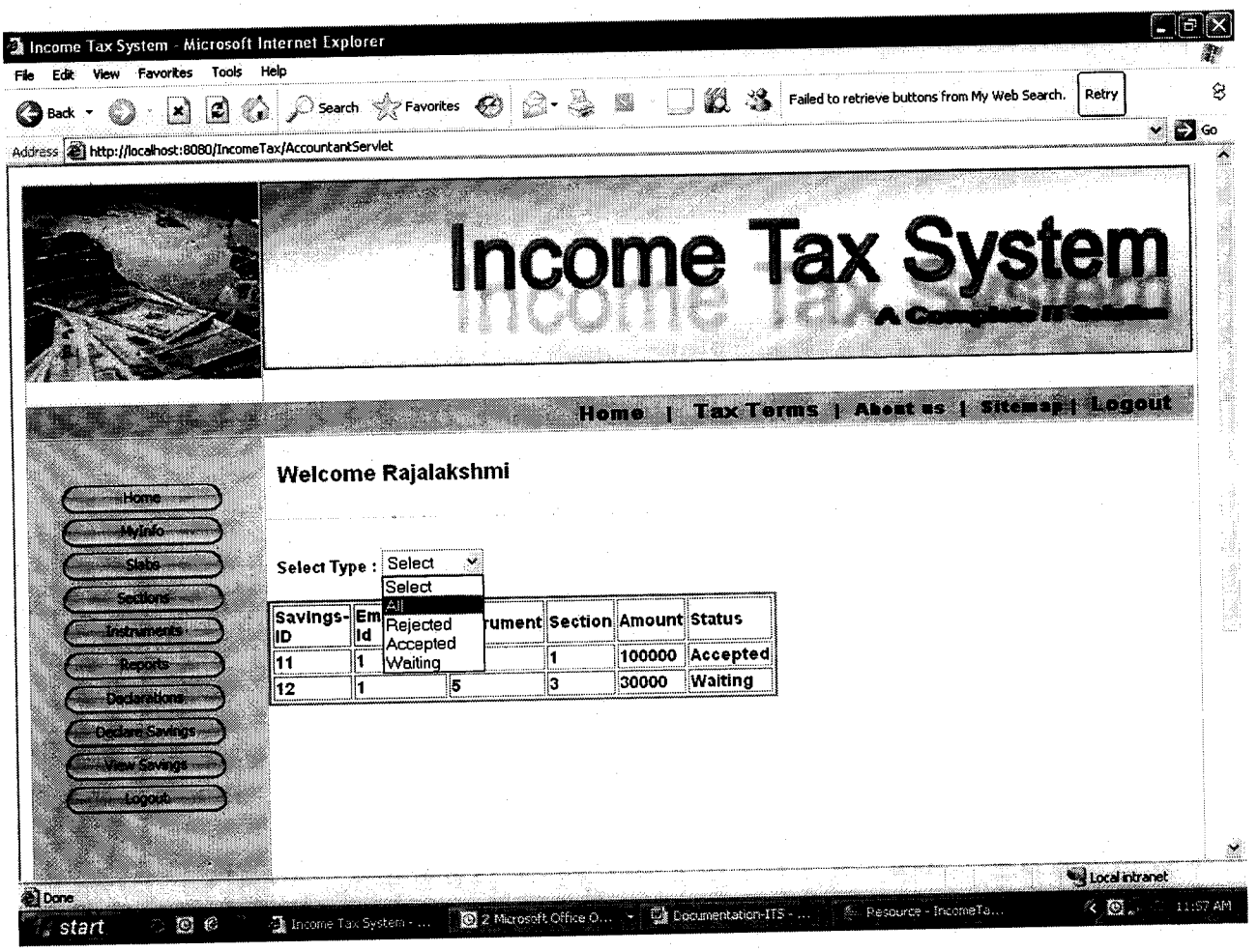


Figure A.16 Report of all employee savings based on Status

REFERENCES

BOOKS

1. Herbert Schildt, 'Java @ Complete Reference', Tata McGraw – Hill Edition, 5th Edition.
2. Dan Haywood, Martin Bond (2001), 'Professional Java Server Programming, J2EE 1.5 Edition', Wrox Press.
3. Bruce Eckel, 'Thinking in Java', Prentice Hall PTR, Second edition.

ONLINE REFERENCES

www.jav.sun.com/j2ee/1.5/download.html

www.java.sun.com/jdbc/connection.html

www.w3Schools.com/jscript/index.html