A

Project Report

On

SecUp

Secure File Uploading System

By

Jigar Shah
Krishna Soni
Rohit Wagh

Guided by

Mr. J. A. Bharadwaj

Submitted to

Department Of Information Technology
K. K. Wagh Institute of Engineering Education and Research
Nashik - 422003

University Of Pune
(Year 2004 – 2005)

DISSERTATION APPROVAL SHEET

A

Project

On

"SecUp- Secure File Uploading System"

Has been successfully completed by

Jigar Shah
Krishna Soni
Rohit Wagh

at

Department Of Information Technology
K. K. Wagh Institute of Engineering Education and Research
Nashik – 422003

University Of Pune.
(Year 2004 – 2005)

Mr. J. A. Bharadwaj **Project Guide** **Department of Information Technology.**	**Prof. G. K. Kharate** **Head** **Department of Information Technology**	**Mr. Nirmal Juthani** **External Project Guide** **NetVigil Software Pvt. Ltd.**

INDEX

Table of Figures

List of Tables

"The art of war teaches us to rely not on the likelihood of the enemy's not coming, but on our own readiness to receive him; not on the chance of his not attacking, but rather on the fact that we have made our position unassailable"

-- *The Art of War,* Sun Tzu

Preface

In this age of universal electronic connectivity, of electronic eavesdropping and electronic fraud, there is indeed no time at which security does not matter. This in turn, has led to a heightened awareness of the need to protect data and resources from disclosure, to guarantee the authenticity and integrity guarantee of data and messages, and to protect systems from network-based attacks. Major change that affected security is the introduction of distributed system and the use of networks and communications facilities for carrying data between terminal user and computer and between computer and computer.

1. INTRODUCTION

1.1. Project Objective

This project aims at providing secure, efficient and reliable file uploading from computers that are in untrusted network to trusted network. It aims at providing distributed and reusable components that can easily integrate into any system with small or no modification. It overcomes security issues that are encountered in traditional two - tier architecture by replacing it with three – tier.

1.2. Problem Description

1.2.1. Problem Analysis and feasibility:

As per the discussion with the people from industry (NetVigil Software Pvt. Ltd.), we analyzed the network architecture of their company; and security problems associated with uploading file from terminals in public domain, like cyber café or home computers to internal company servers. Network Architecture of an Enterprise is as follows:

1.2.2. Problem Associated with File Uploading:

As we can see from network architecture, File is first is uploaded to Web server and administrator with some special privileges can access those file or they are transferred to internal we server through some application. As Web server is in Public domain, it is security threat to any enterprise as file contains some sensitive data. In traditional architecture, authentication and authorization details were stored on Web server, which is again a threat to Enterprise Security.

1.2.3. Network Architecture of an Enterprise:

As shown in figure, an Enterprise consists of one or more central servers. In turn, they connect other departmental servers (e.g. production, marketing, sales). Files that consist of confidential data like sales info, market analysis or technical reports; needs to be transmitted from non-trusted terminal to trusted one through internet.

An Organization will host its web site with some ISP or Web service provider to reduce cost in maintaining web server and security related issues. This Website will host on some web server like Tomcat, for example. This website provides user interface for authentication, file uploading etc. Some script on web server will transfer uploaded file from client terminal to web server, to internal departmental terminals.

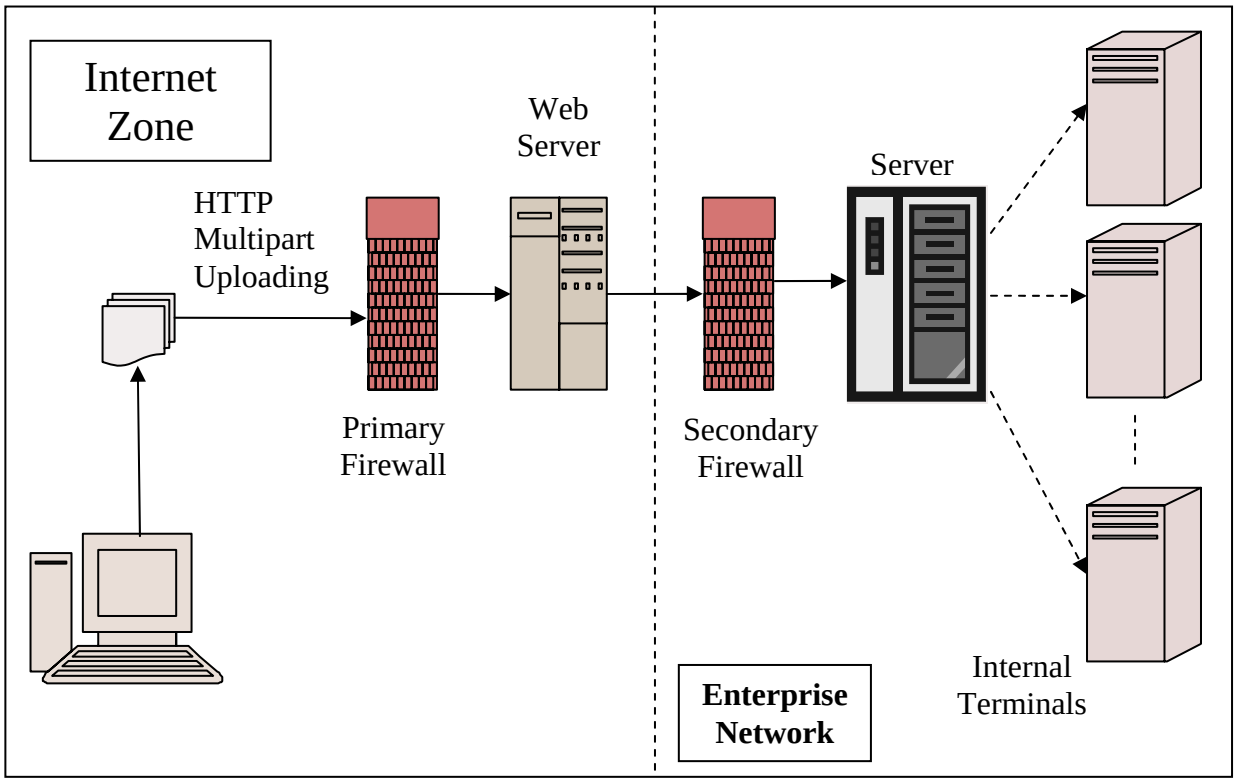


Fig 1.1 Network Architecture of an Enterprise

1.3. Current Implementations

There are many third – party solutions available in market to overcome disadvantages of 2-tier file uploading. They come in ready-to-use packages that can be deployed on different tier with minimum configuration. Various other Off-the-shelf products based on technologies like SFTP, SSH, SSL based VPN are available in the market for solving secure file uploading problem.

1.4. Limitations of current systems

However, these solutions are vendor specific and implemented on proprietary platforms. These solutions lack in interoperability and flexibility. Products based on SSL VPN, SSH increase overhead of encryption and make file uploading extremely inefficient and slow. Additionally they do not solve the security problem of caching file on the web server, they just provide secure channel for file uploading. Similarly, Protocols like SSL, TLS and SFTP make transmission channel secure but packets need to be decrypted at web server to check integrity and other information. Third-party solutions like are platform and vendor specific as they are implemented on platforms (ASP).

1.5. What Can Be Done?

To get rid of security threat of 2-tier file uploading we provide file uploading with 3-tier architecture. As web server is in public domain and we don't want to store files on that, we just bypass this web server and don't allow file to be stored at we server. Additionally store authentication and authorization information on internal server instead of web server; this makes authentication more secured than 2-tiered approach. To implement platform and vendor independent system, use development environment this follows open specification. To provide highly available and extensible file uploading environment, develop a distributed system, which does not rely on any single server and can handle failures of communication link or server efficiently.

1.6. Our project

To implement this SecUp makes uses of platform and vendor independent development platform – J2EE. J2EE is a standard based on Java 2 technology for 3-tierd application development.

Through the implementation of our project, we aim to secure file uploading through following functionalities:

- Authentication module: Authenticate client at internal server side through EJB
- Enterprise module: Distributed components responsible for Business logic
- Web-module: provide user interface through dynamic pages and break file for multipart http upload
- Secure channel for prevent eavesdropping while transmission

1.7. Organization of the report

Section 2 describes the Analysis phase. It contains the project plan and the requirement Analysis. This section also describes the team structure of our project.

Section 3 describes the design phase. The section phase contains the Software Requirement Specification and details about the Risk Management.

Section 4 involves risk analysis and assessment. It measures risk involved in project with respect to resources and project description.

Section 5 involves modeling of the system. It contains the uml diagrammatic representation of the project.

Section 6 details the coding phase. It provides an insight into the programming languages used, platform and tools.

Section 7 concerns the Testing phase of the project. Software testing is critical element quality assurance for design, and coding.

Section 8 discusses the Software Quality Assurance (SQA) Plan.

2. ANALYSIS

2.1. Project Plan

The Project Plan is aimed at estimating the amount of work, resources and the time required to complete the project. The metrics drawn are estimates and will invariably change as time progresses. This exercise will help us to suitably distribute and allocate the work tasks equally among the members based on the time line and the way time allows.

2.2. Software Scope

To make a proper estimation on the software front, the boundaries of the scope of the software needs to be very clearly defined early on in the entire process. This is done in the SRS section that follows in the document. All the following estimates are done with respect to this scope definition.

2.3. Resources

The resources that would be utilized during the project lifecycle are:

2.3.1. Human Resources: The project team is comprised of 3 (three) members as listed below, in alphabetical order:

- Jigar shah
- Krishna Soni
- Rohit Wagh

Officially, the project is allotted 6 hours per week (h/w) over a period of roughly 8 months.
Thus the available time period is:

6 h/w x 3 persons x 32 weeks (w)	=	576 man-hours
	=	34 man-months
		(@17 man-hours/man-month)

2.3.2. Software Resources: Various open source/free software components used are as follows:

- J2EE Web server (Tomcat-Apache)
- J2EE Application Server (SunOne Reference Implementaion 8.1SP4)
- JDK 1.4 or higher
- Pointbase Database server

2.3.3. Environmental Resources: The basic requirements of the development environment are:

2.3.3.1. Hardware Requirements

- 3 Computers, one each for each tier; With following configuration
    - 512 MB RAM or higher (For Application server, other computers may have standard PC specification)
- Ethernet LAN 10/100Mbps, network cards, cables, hubs/switches

2.3.3.2. Software Requirements

- Netbeans 4.1 IDE – J2EE
- Microsoft Windows 2000/XP SP2 or Linux Enterprise Edition
- An Internet domain name (www.companyname.com)

2.4. Project Estimation

We decided to utilize an LOC based estimation method to determine the size of the task and the approximate time required for completion. Approximation of LOC is based on complexity, past project experience and analysis of existing systems. The approximate LOC breakup of the various modules is as follows:

2.4.1. File Uploader :- Web-Module

Sr. No.	Module Name	Approximate LOC
1.	File Uploader Servlet	2500
2.	File Uploader Web-Interface Module	150
3.	File Uploader Application Server Module	500

Table 2-1: File Uploader Web-Module LOC Estimation

2.4.2. Authentication and Authorization:

Sr. No.	Module Name	Approximate LOC
1.	Authentication Web-pages	300
2.	Authentication Web-module	400
3.	Authentication EJB	700

Table 2-2: Authentication and Authorization LOC Estimation

2.5. Make-Buy Decision Table

Sr. No.	Component Name	Discussion	Decision
1.	Authentication Bean	Authenticating and Authorizing user is a vital element of a secured system. As mentioned earlier, Authentication data coming from client will be passed to the internal server for Authentication and Authorization. To protect client from sending password and username in plain text (conventional form based Authentication) we use SSL based authentication.	MAKE, as in conventional form based authentication takes place at web server; which is security threat, SecUp will implement EJB for authenticating client. This EJB will Access User database and send authorization information to the web server.
2.	SSL certificate	This certificate is very important from authentication perspective. Web server and Application server both provide tool for generating certificate but these certificates are easy to forge. Therefore, we need some competent authority to generate and verify authenticity of the certificate.	BUY, therefore we will prefer to buy these certificate from certificate authority like 'Thwate' or 'Verisign'
3.	File Splitting web module	In case of standard HTTP multipart file uploading, File size handling is taken care by web browser only. In case of SecUp we need custom setting for breaking and assembling file part. Additionally, Web server has limitation for file size that can be uploaded. To overcome these problems we have to device our own web component to break file into required size.	MAKE. We will develop Java Servlets for web serer to implement this functionality. We will make modification in a standard o'reilly file uploading library to support multipart file uploading.
4.	File Uploading bean	This is a core module of this project. It is responsible for receiving chunks of data and assembles them. It is also responsible to transfer file to appropriate internal file server based on authorization information provided by authentication bean.	MAKE. This module will be implemented on J2EE specification to allow interoperability without recompiling code. This module will be implemented with distributed component to allow flexibility and scalability.
5.	User Interface Module	This consists of dynamic web pages, which are responsible creating and deleting session. This	MAKE. These web pages need to be custom made to fit requirements

		will provide user interface for both admin and user. This module includes various dynamic web pages for authentication, browsing file on client computer, selecting file destination, adding user.	of our project. These will be made using JSP and java script.

Table 2-3: Make-Buy Decision Table

2.6. Project Scheduling

Sr. No.	Period	Work to be done
1.	1st Jan 2005 to 10th Jan 2005	Basic Study of Requirements.
2.	5th Jan 2005 to 15th Jan 2005	Research Work in parallel with Requirement Study.
3.	15th Jan 2005 to 25th Jan 2005	Design and Analysis.
4.	25th Jan 2005 to 5th Feb 2005	Coding
5.	5th Feb 2005 to 20th Feb 2005	Testing

Table 2-4: Project Schedules

2.7. Requirements Analysis

This section is intended to provide a concise but complete description of the requirements of the project. This section

2.7.1. Login

- Visibility of buttons and form fields
- Error description while typing password (no special symbol, numbers as beginning of password)
- Report error when password and/or login field is empty

2.7.2. Administration

- Add user – Select department, provide initial password
- View logs

2.7.3. User/Administrator

- Login
- Changing Password
- Uploading File
- Logout

2.8. Requirements Elicitation

This section describes requirements of three tiers.

2.8.1. Web module:

Web module will reside in DMZ zone of the network. This is semi-secured zone. These modules is responsible for breaking file into parts and send authentication information and file parts to internal web server. This module will be deployed on J2EE web server and will host dynamic web pages. Following functionality will be implemented in these module

- File Upload
  - Invoke Enterprise Java Bean for File uploading.
  - Brea k file parts and stream all the parts
  - Provide file parts and other information to file uploading bean
- Authentication
  - Get Authentication and Authorization details from client and pass it to internal server
  - Provide dynamic web page for browsing file on client machine
  - Add user to specific department

2.8.2. Enterprise module:

This module will reside on internal network (application server). This module is responsible for performing various server functionality.

- File Upload
  - Assemble file parts received by the server
  - Check for the duplicate file name on destination machine
  - Check for the user authorization
  - Redirect file to appropriate internal file server
- Authentication
  - Check for the authentication of the user by comparing data received from web server and data from the database.
  - Give appropriate information to web server and File uploading module
  - Update user database with changed password
  - Add user to internal database
- Activity Logging
  - Keep log of error messages on application server and web server
  - Track user activity like login & logout time, file uploaded

2.9. Requirements Study And Classification

Client

- **Necessary Requirements**
  - Java enabled Web browser
- **Desirable Requirements**
  - None
- **Optional Requirements**
  - None

Web Server module

- **Necessary Requirements**
  - J2EE compliant Web server
  - Support for generating SSL certificate
  - Firewall software
- **Desirable Requirements**
  - Firewall support for blocking a specific port or service
- **Optional Requirements**
  - Support for crash recovery and clustering

Application Server module

- **Necessary Requirements**
  - J2EE compliant
- **Desirable Requirements**
  - Support for IDE
- **Optional Requirements**
  - Support for Clustering

Authentication and Activity tracking Module

- **Necessary Requirements**
  - J2EE compliant Database driver
- **Desirable Requirements**
  - Transaction Management support
- **Optional Requirements**
  - None

2.10. Team Structure

Team Members

The members of the project team are as follows:

- Jigar Shah
- Krishna Soni
- Rohit Wagh

Internal Guide

The Internal Guide for the entire project:

- Prof. J.K. Bhardwaj

External Guide

- Mr. Nirmal Juthani – (NetVigil Software Pvt. Ltd.)
- Mr. Dhaval M. Shah

3. DESIGN

3.1 Software Requirement Specification

3.1.1 Introduction

3.1.1.1 Overall Description

Due to Globalization, major organizations today have several branches distributed globally. With the technological advancements, any authorized employee of an organization can communicate with any branch across the world and from anywhere using Internet. One of the major tasks that he performs is the transfer of files to the actual destination m/c, which is the part of an organization's internal network. The critical phase during the above operation is the uploading of a file onto the server. This operation is done using the Web Server as an intermediate file caching system. This makes it unsecured as the Web Server resides in the public domain and can be accessed by anyone.

The proposed system should provide security throughout the operation of uploading the file on the internal server. The web server should not reveal the file to any unauthorized user even after it is attacked. It should also support multiple connections with the clients and should also work with multiple platforms.

3.1.1.2 System Constraints

- System will not sustain hardware failures on the client side.
- Connection failures due to problems in the connection or Internet will also not be handled.
- System will not be responsible for any attacks made to the internal server of the organization due to absence of or attack on the firewall.
- Security threats on the client side revealing confidential details may also lead to problems which will not be handled by the system.

3.1.2 Information Description

3.1.2.1 Hardware Interfaces

The proposed system will require web servers and application servers. The Application Servers will be a part of the Intranet and will be connected to the internal servers of the organization. The Application servers will also be connected to the Web Servers, a part of the Internet with possibly a firewall in between.

3.1.2.2 Software Interfaces

The software interfaces includes operating systems, web browsers on the clients, web servers and Application servers. It will also require an interface on the Application server side for multiple connections from the Web servers.

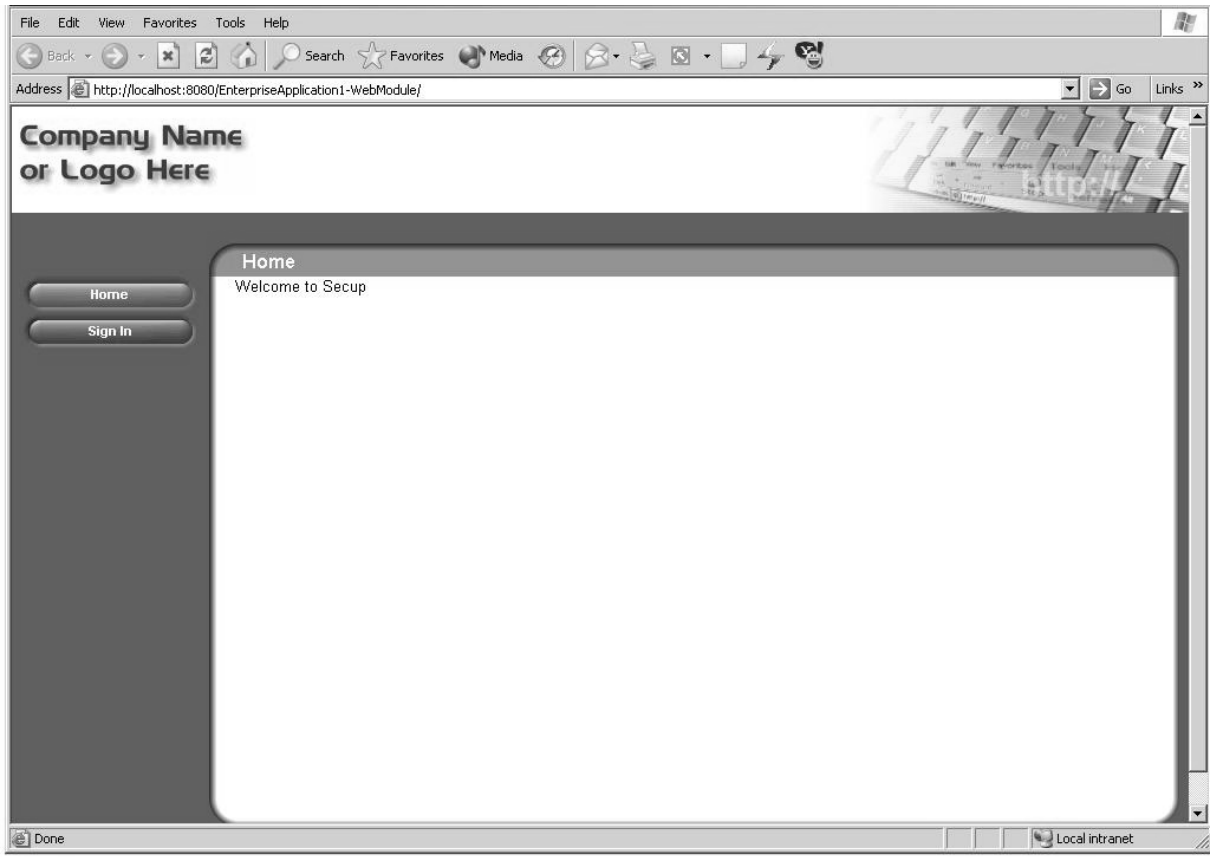
3.1.2.3 User Interfaces

Description of the HCI

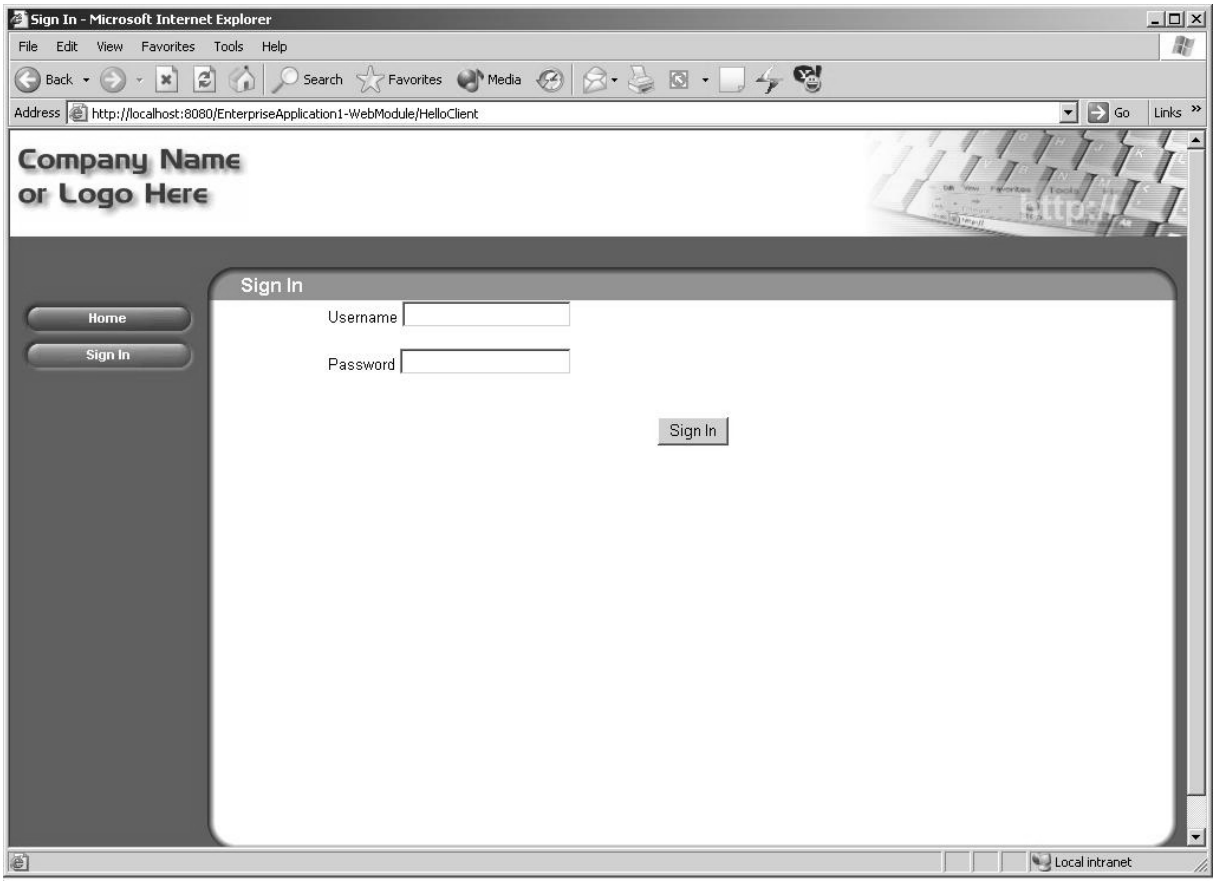
SecUp has separate interfaces for users and administrator. The administrator can update the details of the users, upload files and view the logs.
Users are provided interfaces for uploading files and changing their login information.

Prototype User Interface Screens

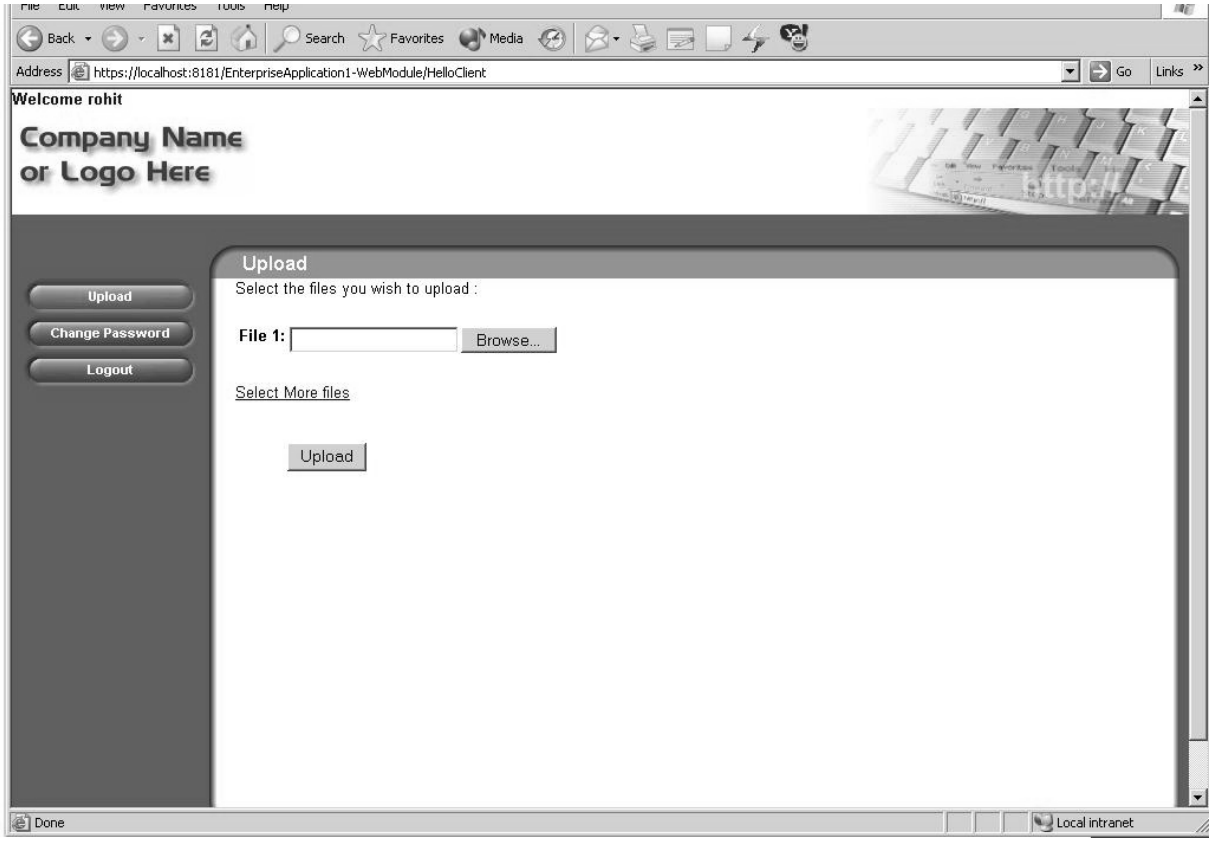
- **Home Page**



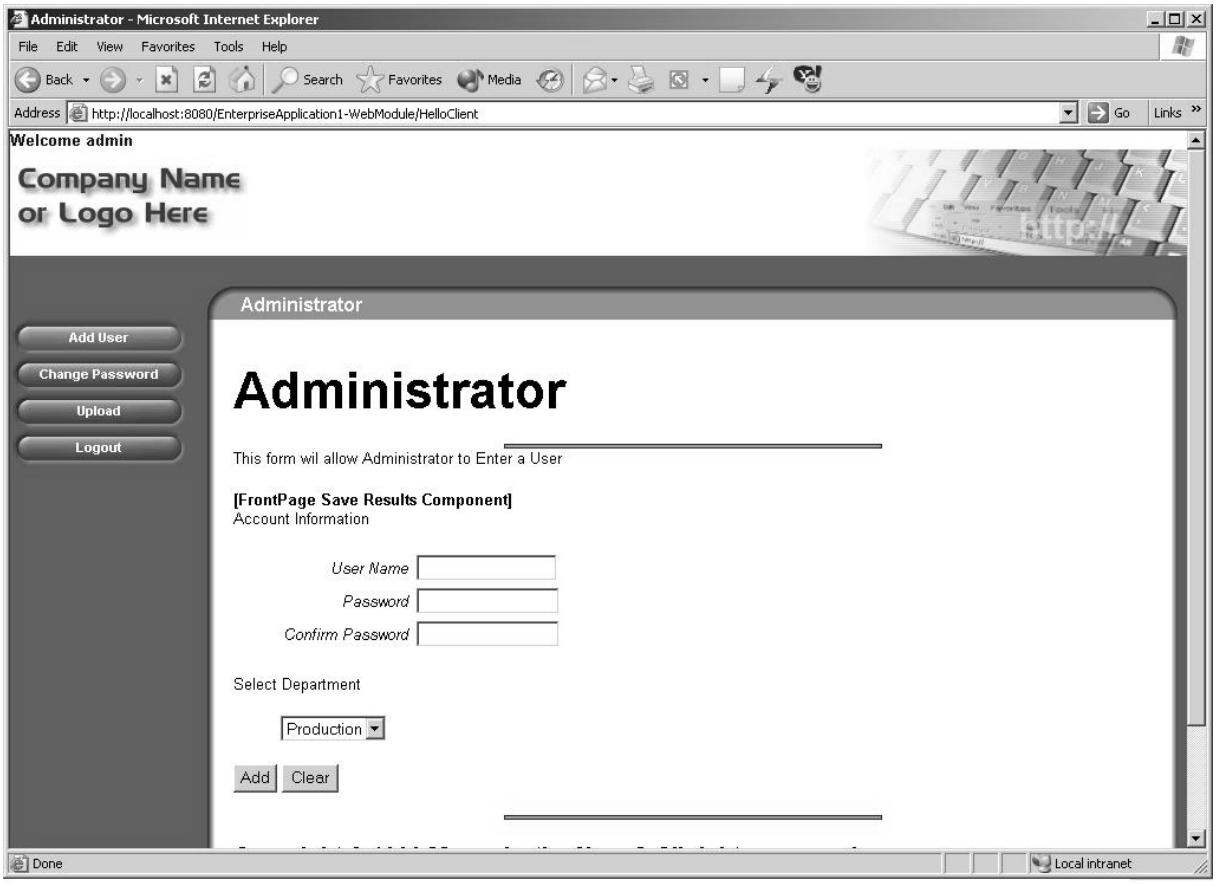
Sign In Page:



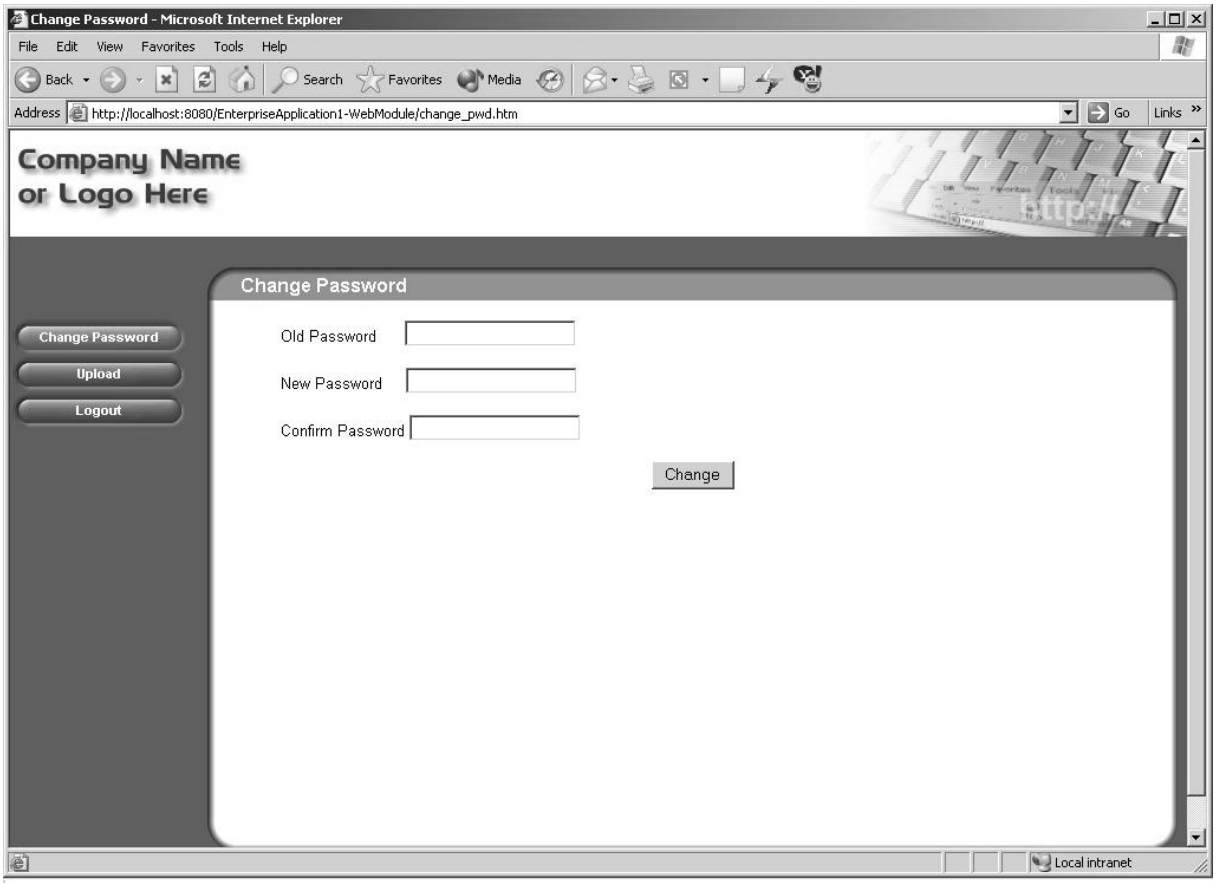
User's Home Page:



Administrator's Home Page:



Change Password Page:



3.2 Functional Description

3.2.1 Functional Partitioning

It consists of **four** modules:

Module 1: User Interface
Module 2: Authentication and Authorization component
Module 3: Multipart File Uploading Servlet
Module 4: File Uploading Components on Application server

3.2.2 Functional Description

Module 1: User Interface:

This module consists of web pages for user interaction. These Pages are on Java Web Server. SecUp uses Sun One as a Web Server. It is an open Source java Web server.

Module 2: Authentication and Authorization:

This module authenticates users using Pointbase Database which resides on the Application Server. User provides username and password as the login information and is redirected to the appropriate page after validation.

Module 3: Multipart File Uploading Servlet

This resides on the Web server. It performs multipart file uploading. This is a critical module, as it is responsible for not allowing any part of data to be temporarily stored on to the disk. It reads the file contents send by the client and buffers them, which in turn, invokes EJB component residing on Application Server. The components are located using JNDI lookup.

Module 4: File Uploading Components on Application server

This is the heart of SecUp. These are EJB components. Inherent security and JNDI lookup feature makes EJB components truly location independent. Servlet instantiates EJB when its buffer is full and passes the buffer to the component. EJB then writes the buffer to a file and stores this file on appropriate internal file servers mapped on the Application Server. The information of the file servers is again retrieved from the pointbase database. The result is then returned to the Servlet, which in turn informs the client giving him necessary information.

3.2.3 Supporting Diagrams

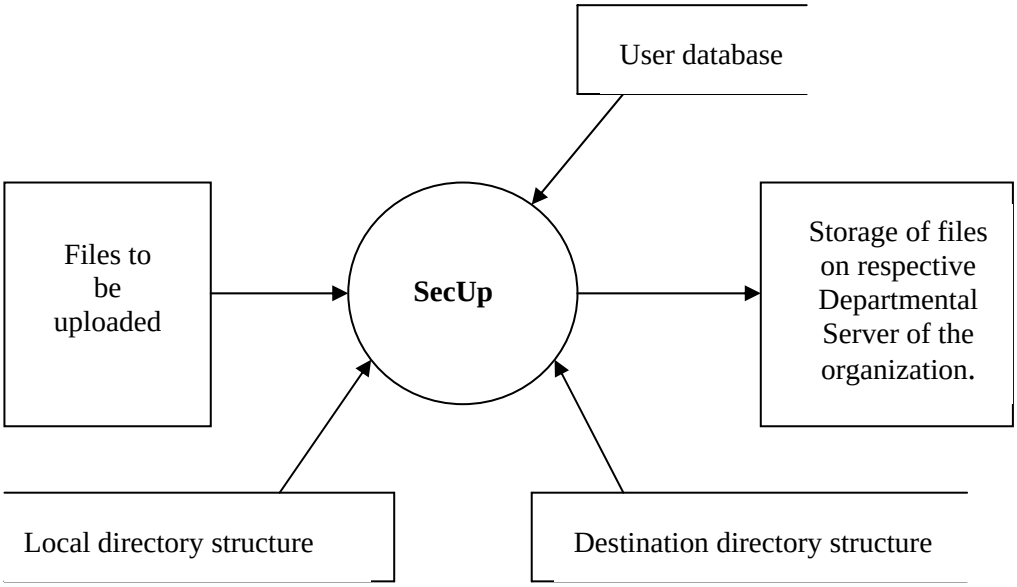


fig.3.1 Level 0 DFD for SecUp

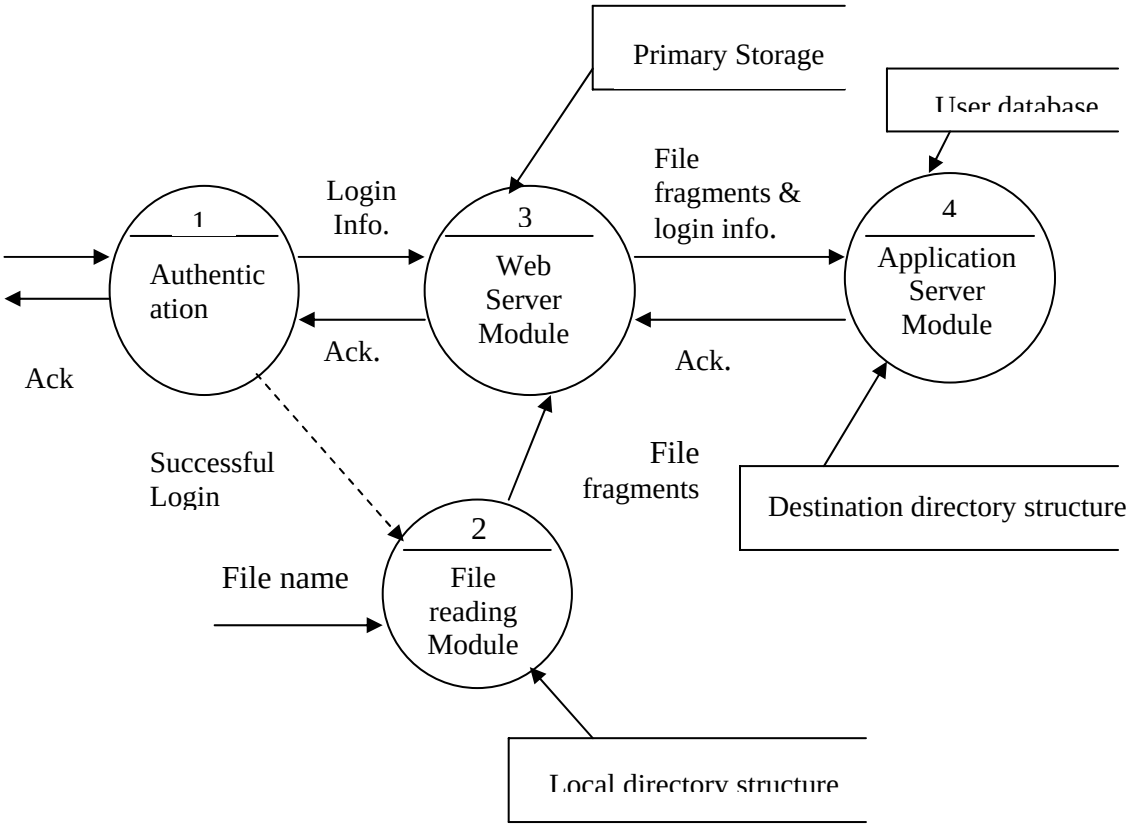


Fig 3.2: - Level 1 DFD for SecUp

`

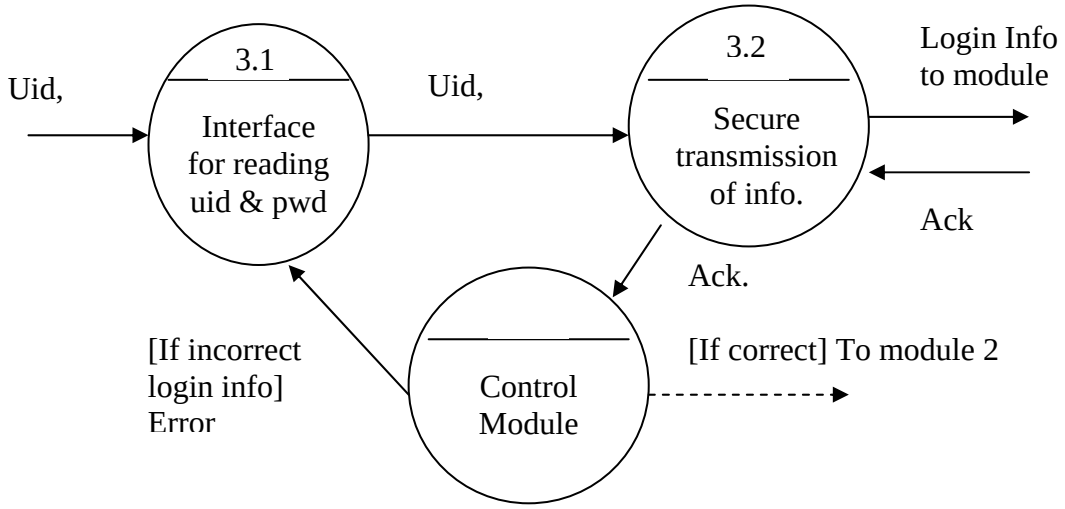


Fig 3.3: - Level 2 DFD for Authentication (1)

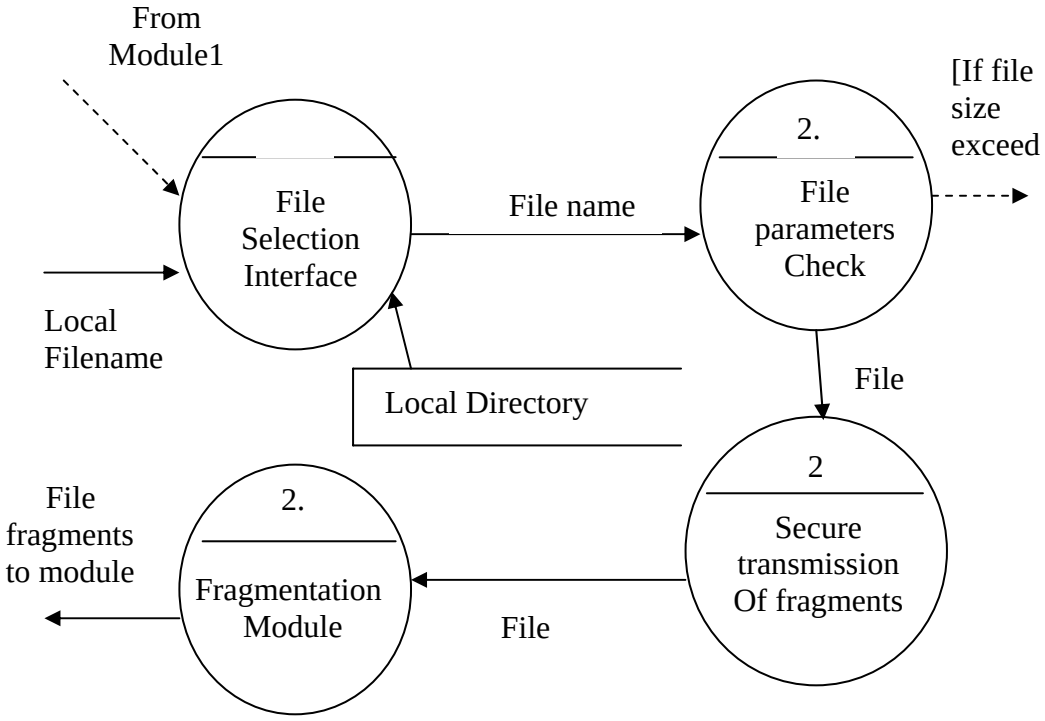


Fig 3.4: - Level 2 DFD for File reading Module (2)

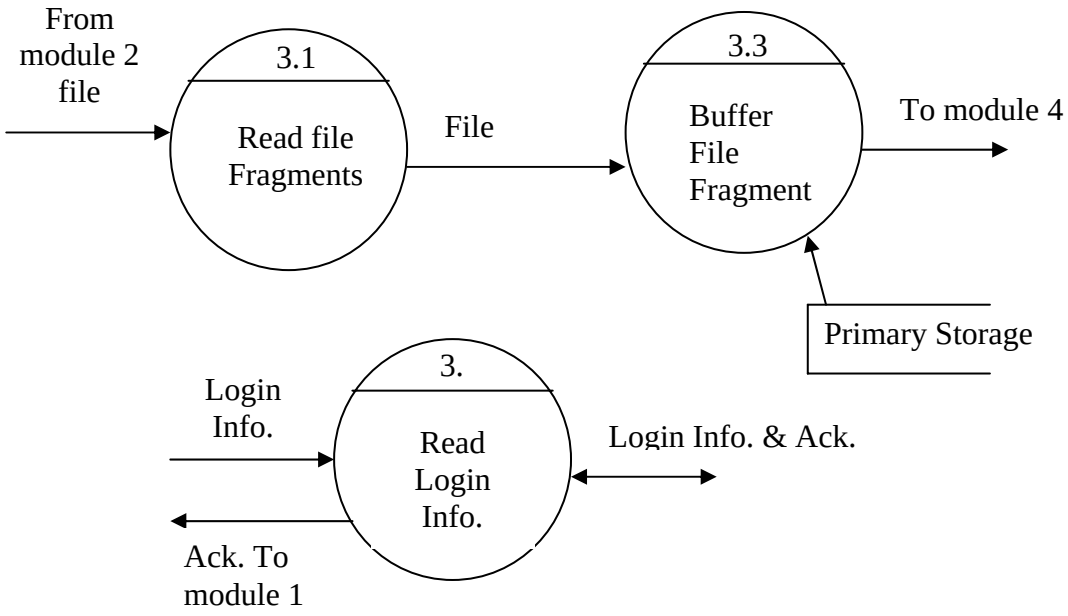


Fig 3.5: - Level 2 DFD for Web server module (3)

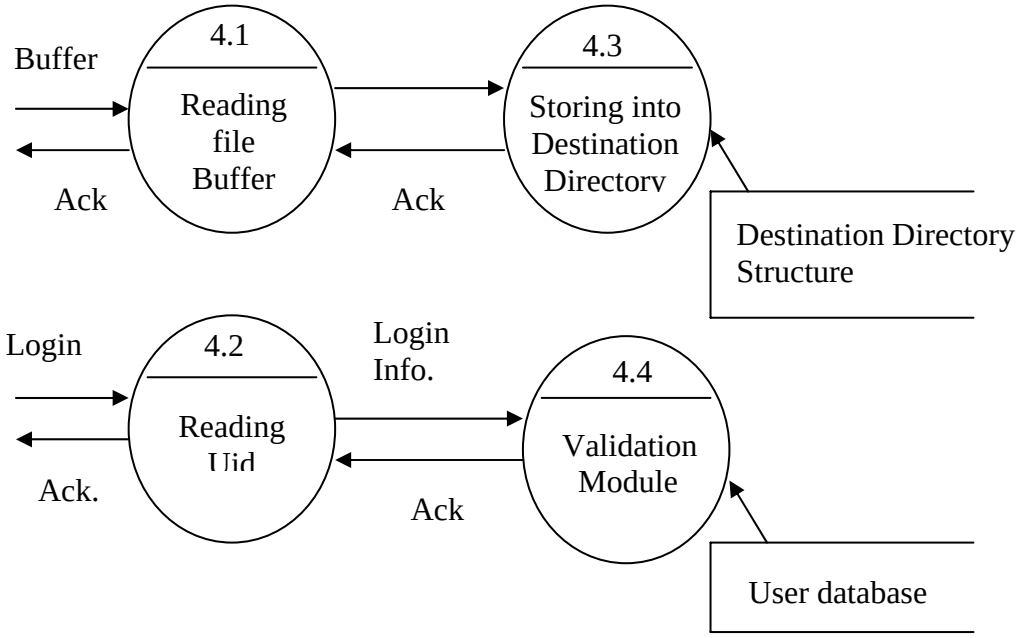


Fig 3.6: - Level 2 DFD for Application Server Module (4)

3.3 Behavioral Description

- On getting the initial request for the homepage from the client browser to the web server, the home page is sent back to the client. The client then can request for signing in. A SSL session is instantiated as soon as this link is accessed. A certificate is sent back to the client proving the server's identity.

- The web server sends back the sign up page to the client browser. The client now can provide the username and the password for proceeding further. This information is secured by the session. After signing in the user is redirected to his homepage, where he can upload a file or change his login information.

- The username and the password is checked on the application server and the user is redirected to the next page. In case the login information is incorrect, the user is not allowed to move further.

- User can be an employee or and administrator. SecUp takes care of redirecting the user to his privileged area.

- Administrator can add users giving information like username, password and their department. In case the user already exists, this user is not added and appropriate message is sent to the administrator. Administrator is also allowed to upload a file to his file server.

- User can upload any file residing on the client machine. They need to provide the path and the filename and click the upload button. A maximum of 10 files can be uploaded simultaneously. After the file is successfully saved on the respective file server the user is acknowledged by giving him the necessary feedback.

- Users can also change their login details by providing the old details which are checked and the new details are updated on the database.

3.4 Performance Bounds and Expected Software Response

Module 1: User Interface

The performance of the system depends on the network traffic at that moment. The page might not be displayed when network traffic is high or might get very slow. In case the web server fails and there is no other backup server then the client request might not be fulfilled.
On providing the URL the correct homepage should be sent to the client by the web server. The formatting of elements on the page should be proper and uniform. All the links on the homepage should redirect the client to the appropriate page. No dead links should exist.

Module 2: Authentication and Authorization component

The performance of the system depends not only on the network traffic between the client and the web server but also on the link between the web server and the application server.
On providing the correct login details the user should be redirected to the appropriate page. In case the login details are incorrect then the user should not be allowed to proceed and should be notified by giving appropriate error message.

Module 3: Multipart File Uploading Servlet

In case of heavy n/w traffic or load on the web server the file uploading part might get very slow. If the file size is large it may again take a long time getting uploaded. Any problem on the application server side might also create a bottleneck in the performance of the system.
The incoming file should be buffered properly into the primary memory of the web server. Appropriate application server should be contacted for final storage of the file.

Module 4: File Uploading Components on Application server

The performance of this module depends on the load on the application server and the hardware configuration of the server. It also depends on the performance of Intranet components.
The file contents should be read properly from the buffer sent by the servlet and appropriate file should be created on the right file server. The database should be referred for retrieving the destination of the file based on the user sending it.

4. Risk Assessment

- Risks have been identified and categorized as:
- Project Risks
- Technical Risks
- Business Risks
- The probability values indicates the likelihood of the risk becoming a major issue
- Impact values have been assigned to each level as per the gradation of severity of risk
- Catastrophic – 1
- Critical – 2
- Marginal – 3
- Negligible – 4
- Based on the probability of occurrence and the impact, suitable mitigation plan/action has been taken

4.1 Project Risks

Sr. No.	Risk	Probability	Impact	Mitigation plan/action
1.	Failure of initial estimation of LOC could exert extreme pressure on the schedule and the delay could extend the project completion deadline substantially	0.6	2	Schedule has been prepared such that the project should under the current estimation be completed well before the actual deadline
2.	Requirements may change as newer developments/findings surface. This unstable scope could take the project heavily off schedule.	0.5	2	Maximum stress will be given to obtain the final requirements before the end of the design phase
3.	The estimated time for learning the technology and necessary tools may exceed the set goals, could lead to schedule non-conformance	0.4	2	The learning process will be started before time during the first semester to get an idea of the time required so that later past experience can forewarn us of such a possibility
4.	The complexity of the project has not been judged specifically well, leading to extreme stresses on	0.3	2	Such discrepancies will be detected during the first semester before the final design phase

	schedule and changes			
6.	A project member may fall ill leading to the loss of one person for a prolonged duration of time	0.5	2	Other members of the team should be kept abreast of other member's work, so that one of them can take over in such and emergency. The delay will have to be suffered.
7.	High semester loads may lead to lesser time devoted to the project than promised.	0.3	2	Maximum utilization of the vacations.

4.2 Technical Risks

Sr. No.	Risk	Probability	Impact	Mitigation plan/action
1.	A good network for testing may not be available at the college.	0.6	1	College network is already available for testing. In the worst-case scenario, a separate network should be set up.
2.	Proper O.S. not installed on the system.	0.5	3	Ensure proper installation and O.S. requirements well before implementation.
3.	DNS not working as per requirements.	0.3	2	There may be some problem with the protocol of the O.S. of the n/w present. Either change the protocol or re-setup the network.
4.	Web Server not present at remote PCs.	0.2	3	The network requirements specifically state that the system should have a Web Server installed on it.
5.	Application Server not installed.	0.1	2	It given in the documentation that it should be installed first.
6.	Heavy network traffic in the Intranet	0.1	1	The network has to be checked and proper changes should be made to the network.
7.	Traffic On the Internet.	0.6	1	User has to suffer delays during such peak hours.
8.	Real World scenario is difficult to test in the college network because of the absence of Web Server.	0.4	2	It is fine for testing purpose but some other network might be used to test actual performance.

4.3 Business Risks

Sr. No.	Risk	Probability	Impact	Mitigation plan/action
1.	The application server and the web server have high system requirements in terms of hardware.	0.1	1	The existing system can be upgraded for these requirements.
2.	The web-server or the application server is not compatible with the O.S. used	0.2	2	Either the O.S. should be changed or the correct version of the servers should be installed.
3.	The end user will have to bear the Internet charges when surfing from home or cyber-café in terms of Internet usage.	0.1	4	The User could use a plan/tariff rate that is economical in terms of Internet usage.

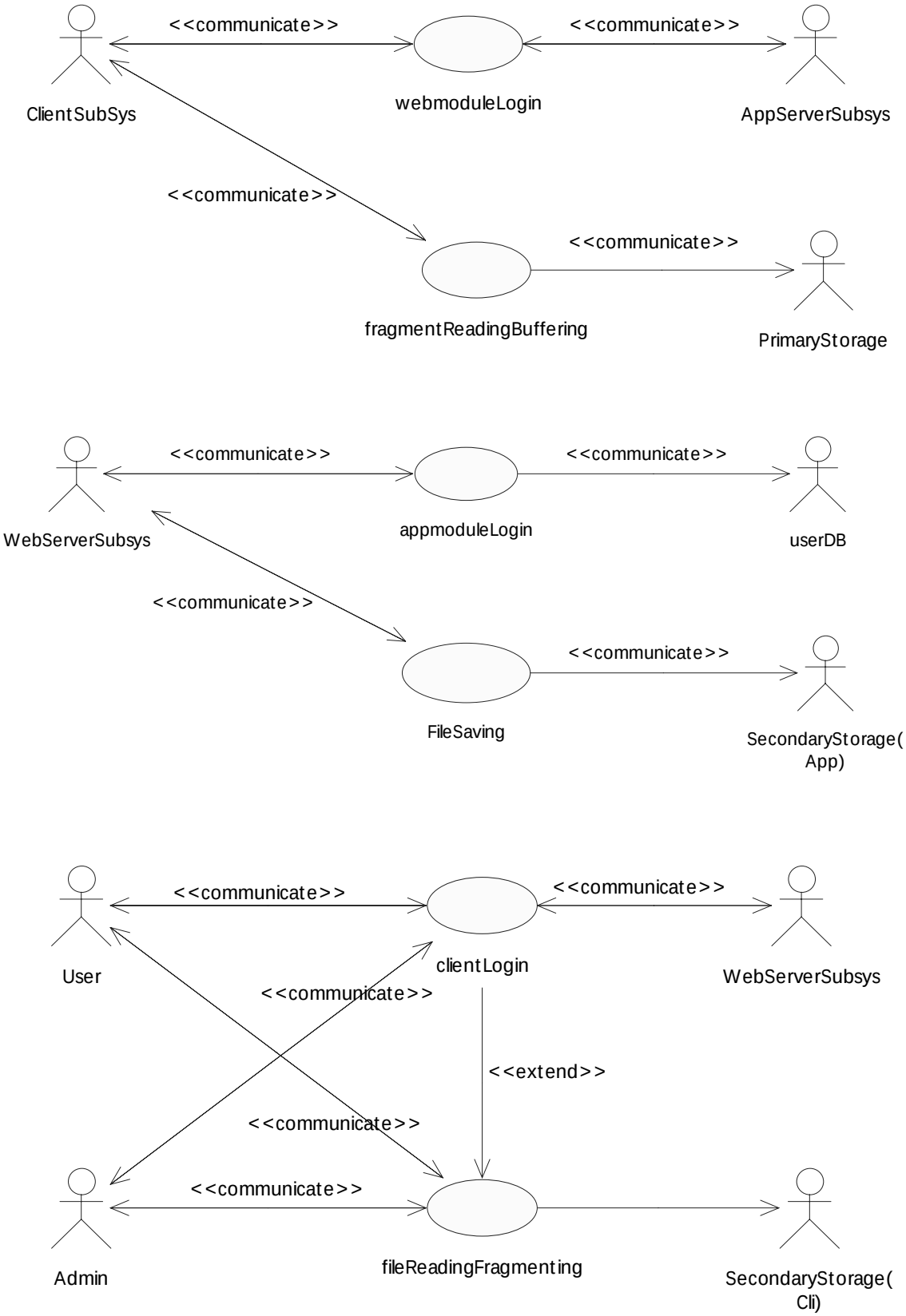
5. MODELLING

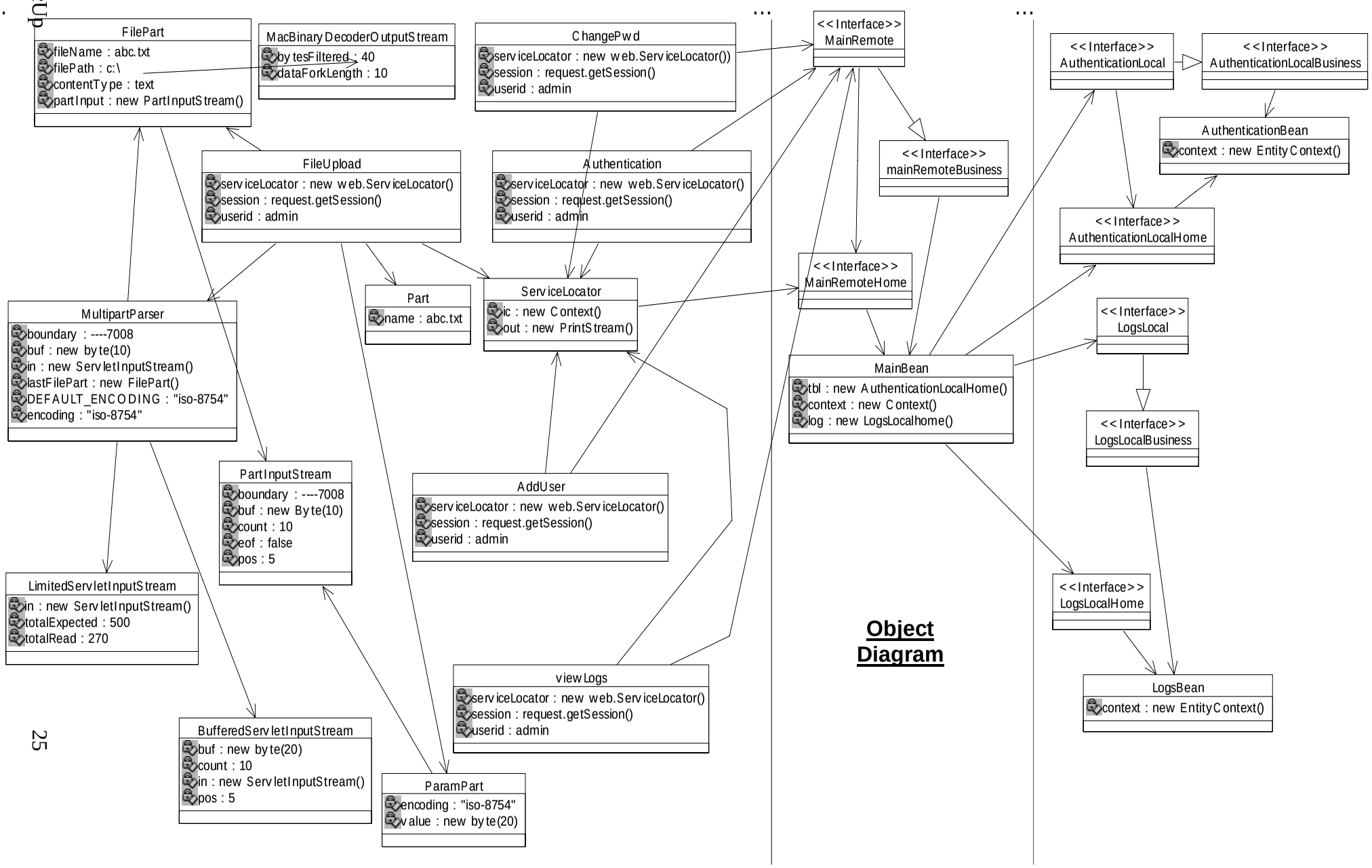
5.1 UML Diagrams

This section consists of the UML Diagrams drawn during the design phase as listed below:

- **Use Case Diagram**
- **Class Diagram**
  (On sheet attached overleaf)
- **Object Diagram**
- **Sequence Diagram**
  (On sheet attached overleaf)
- **Collaboration Diagram**
- **State - Chart Diagram**
- **Activity Diagram**
  (On sheet attached overleaf)
- **Component Diagram**
- **Deployment Diagram**

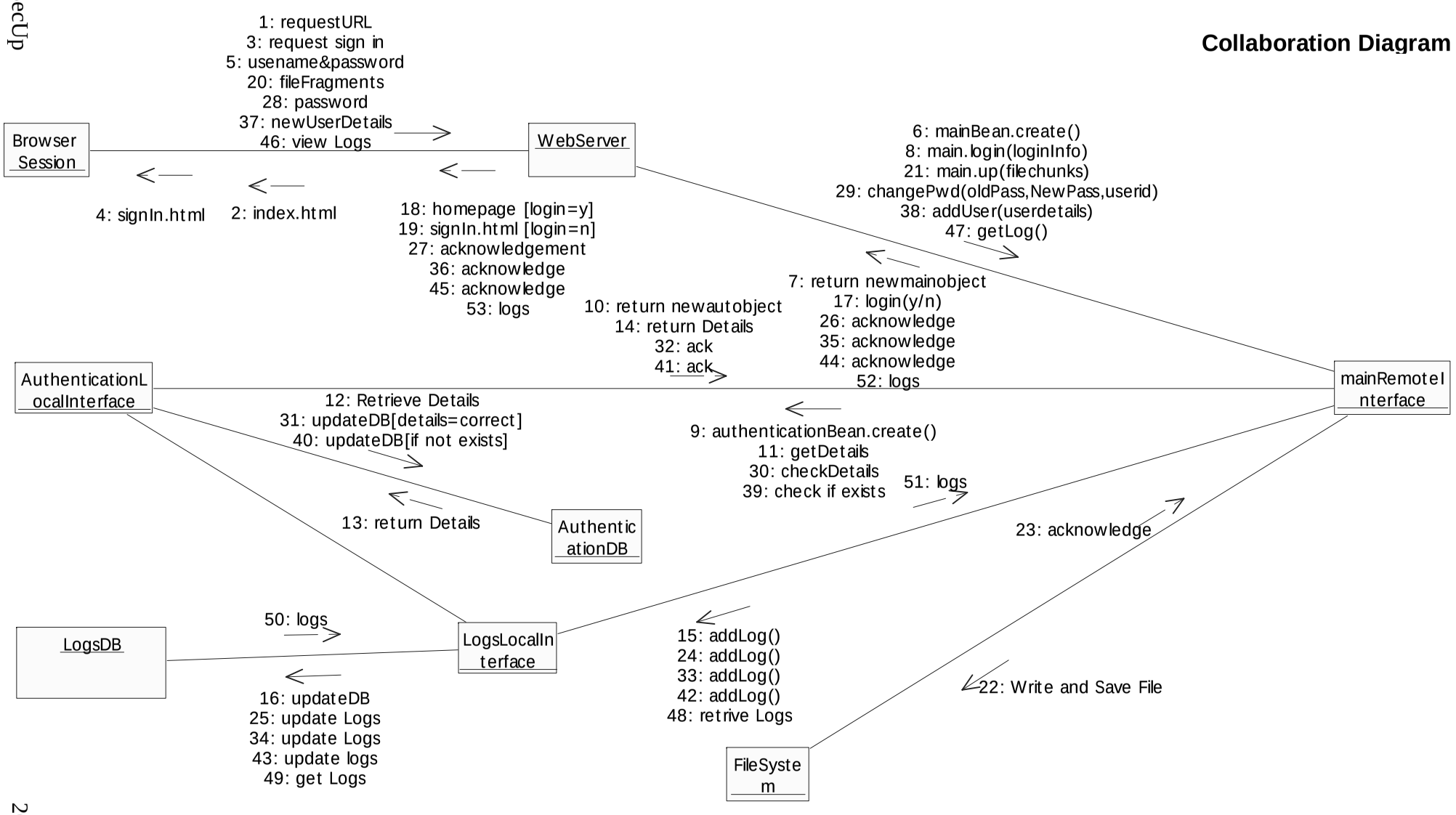
Use Case Diagram

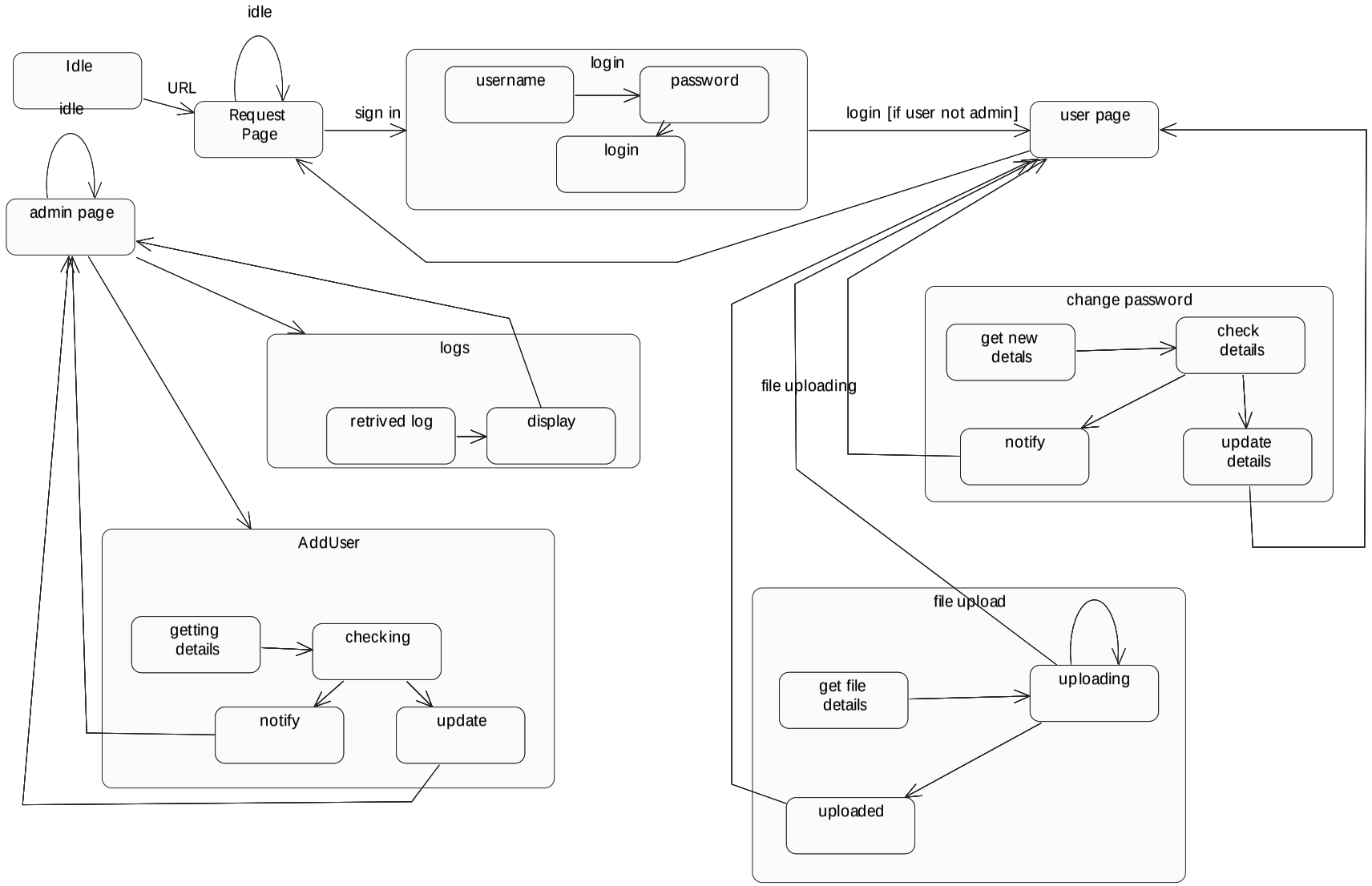




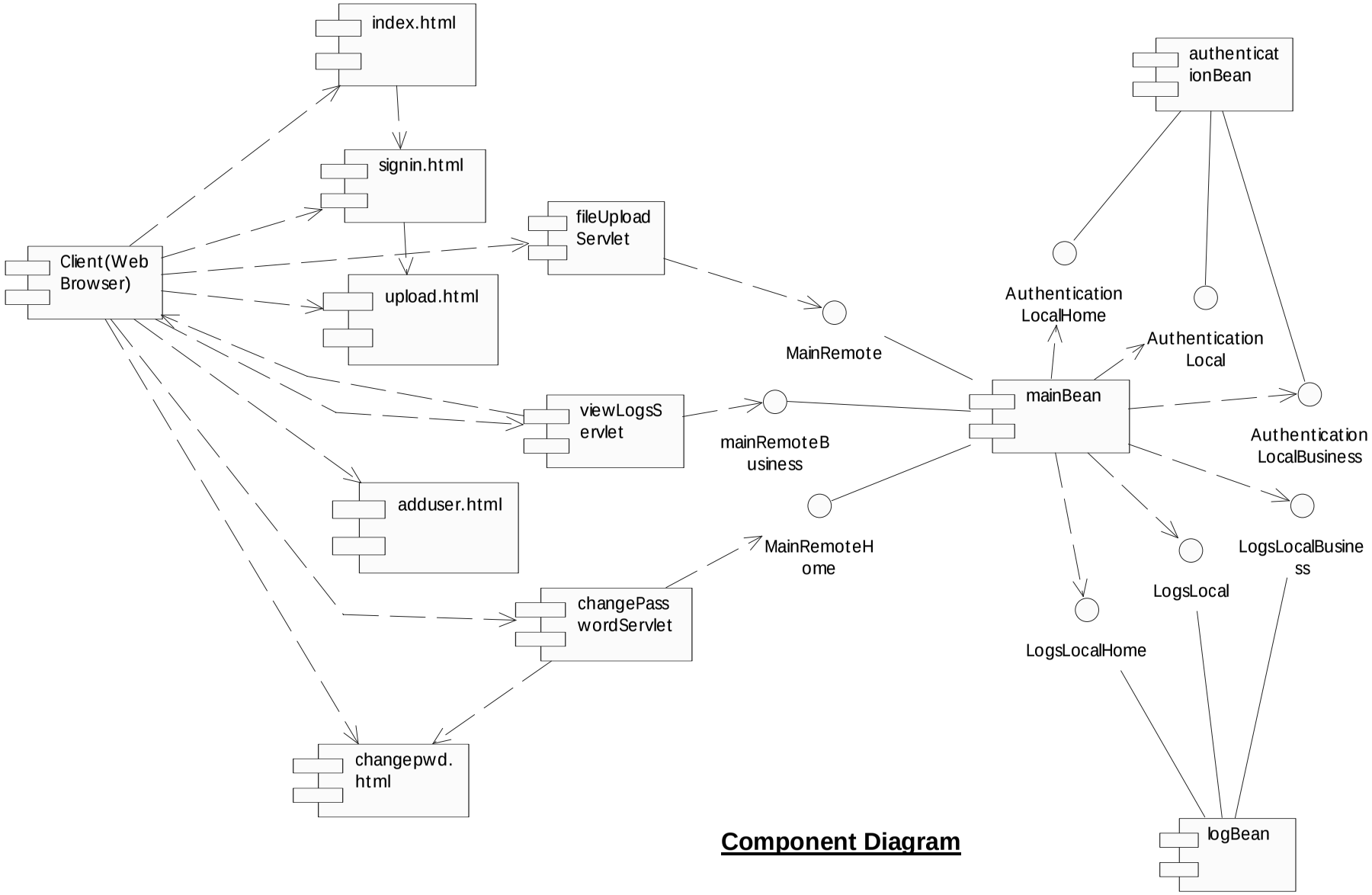
Object Diagram

Collaboration Diagram

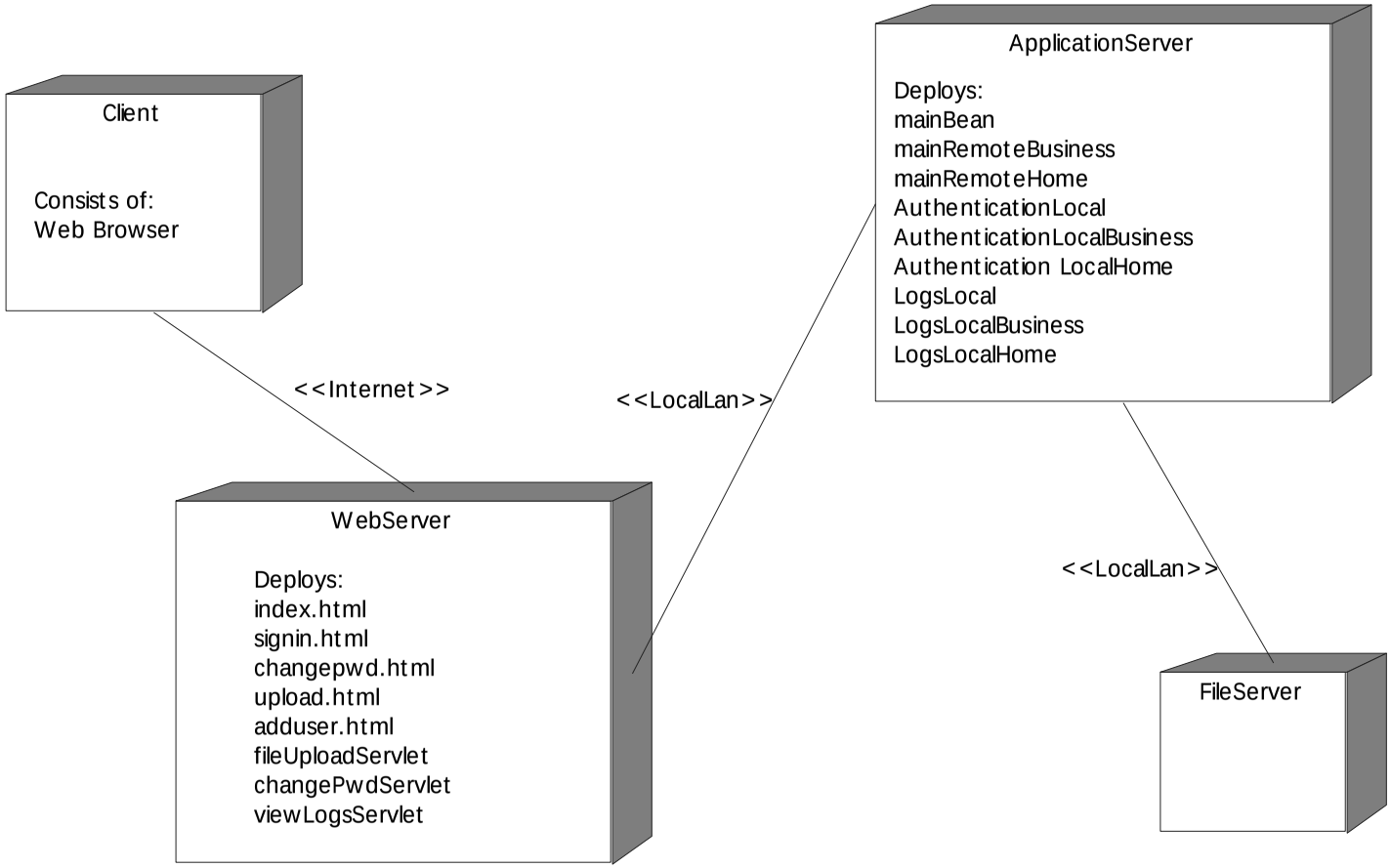




State - Chart Diagram



Component Diagram



Deployment Diagram

6. CODING

6.1 Software Used

- NetBeans 4.1 IDE
- Rational Rose 2000
- Sun One Application Server
- Sun One Web server
- Sun One pointbase

6.2 Hardware Specifications

6.2.1 Web Server and Application Server

- Intel Pentium IV 1.8 GHz
- 256 MB RAM
- 200 MB Hard Disk Space
- Recommended System Requirements:
- Intel Pentium IV 2.0 GHz
- 512 MB RAM
- 400MB Hard Disk Space

6.2.2 Client Systems

- Intel Pentium II 1.8 GHz
- 64 MB RAM
- 56kbps Data Modem
- Recommended System Requirements:
- Intel Pentium III 2.0 GHz
- 128 MB RAM
- Broadband Internet Connection

6.3 Programming Language

NetBeans 4.1 has been used for coding purpose. Microsoft Interdev has been used for developing the User Interface.

6.4 Platform

Platform Independent – J2EE

6.5 Components

- JDBC drivers for Pointbase database
- Stateless Session Beans 2.1
- Entity Beans

6.6 Coding Style Followed

- Necessary Indentation
- Meaningful Variable Names
- Meaningful yet concise comments
- Modular Design
- Low Coupling and High Cohesion

7. TESTING

7.1 Formal Testing Reviews

Formal Technical reviews were done with the following members in the review team:

- Rohit Wagh
- Jigar Shah
- Krishna Soni

The team met regularly to review the progress of themselves and other team members.

In each of the FTRs, the guidelines followed were:

- The meeting was planned well in advance and all team members were intimated
- All details of the component, that were ready, were presented to the team by the developers
- The component was then discussed in the details, as regarding its design and implementation. The working of the component was checked and conformity to the requirements was tested
- The team took one of the three decisions regarding the component based on the review:
- Accept the component as is without any change or modification
- Accept the component with a few changes or modifications
- Reject the component completely and re-implement it
- The review suggestions and decisions were documented
- In each FTR, the points put up during the previous FTR were followed up

The FTRs thus conducted were an umbrella activity and were a part of the Software Quality Assurance (SQA) Plan.

7.2 Test Plan

These are some of the steps followed by the project members in order to test the modules developed:

- Test cases were designed long before the coding and modules were tested accordingly
- The Software Programmer who develops the module does the unit testing along with the other testing member assigned to the module
- Regression testing is done at the module level
- Integration testing is performed after merging of any two or more modules
- Periodic testing of the completed modules with simultaneous white box testing was done

7.3 Unit Testing

Unit testing, done modularly is as follows:

7.3.1 User Interface Module

Test. No.	Test Case	Expected outcome	Test Result
1.	The interface is consistent for all the pages	Correct placement of form fields and their visibility	Ok.
2.	All links are proper.	No dead links	No. Few links were dead and were corrected.
3.	Tab sequence followed.	Tab sequence according to relevance of the input fields	Ok.
4.	Password field kept secret.	Characters should be masked	Ok.
5.	Proper Error messages displayed	Should convey error to user with text and error number	No. Few errors had semantic ambiguity. Corrected.

Table 7-1: User Interface Module: Unit Testing

7.3.2 Authentication and Authorization Module

Test No.	Test	Expected outcome	Test Result
1.	Does it display the correct page if login is correct	Administrator and user should get personal pages with respective login	Yes
2.	Does it prompt for errors if Connection fails due to some reasons?	http error messeges should be displayed with their number and description	Yes
3.	Does it give Invalid login error if username is correct and password is incorrect?	In case of login failure notification page should be displayed	Yes
4.	Does it give User not found error if user does not exist.	In case of login failure notification page should be displayed	Yes

Table 7-2: Authentication and Authorization Module: Unit Testing

7.3.3 Multipart File Uploading Servlet Module

Test No.	Test	Expected outcome	Test Result
1.	Correct reception of complete file.	Show message "successful upload"	Ok
2.	Correct buffering of the file into memory.	No error messages	Ok
3.	Does it invoke the correct EJB?	Successful object creation	No. n/w problem. Corrected.
4.	Does it send the buffer contents correctly to the application server?	File content are same (no CRC error)	Ok
5.	Does it read the parts correctly in case of multiple file uploads.	Show messages "successful upload" for all files	Ok
6.	Does it provide appropriate feedback to the client when file is saved or some error occurs?	No Web server exception messages	Ok

Table 7.3: Multipart File Uploading Servlet Module : Unit Testing

7.3.4 File Uploading Components on Application Server

Test No.	Test	Expected outcome	Test Result
1.	Database server is started?	No exception or error messages displayed	Ok
2.	Does the file get stored on the file server which is mapped on the application server	Error messages "file reading or saving error"	Ok
3.	Does the module return correct information to the web server	Notification to web server about successful operation	No. Formatting is improper. Corrected
5.	Does the system perform according to the file size and network characteristics	Consistent timing for different file size	Ok
6.	Does the session bean call the correct entity bean?	No "bean not found" error message	Ok

Table 7.4: File Uploading Components on Application Server

7.4 Integration Testing

During the integration of the individual modules, thorough testing has been done to ensure that the individual components still perform as expected in the integrated state.

8. SOFTWARE QUALITY ASSURANCE PLAN

8.1 Purpose Of Plan

- SQA can be defined as conformance to explicitly stated functional and performance requirements, explicitly documented development standards and implicit characteristics that are expected of all professionally developed software.
- The role of SQA is to assure the management that the software development work is being performed the way it is supposed to be. Quality management approaches, effective software engineering technologies, testing strategies are some of the important activities of SQA.
- SQA plan specifies the goals, the SQA tasks to be performed, the standard against which the development work is to be measured and the procedures and the organizational structures to be used in each development and maintenance project.

8.2 Team Organization

Post → **Work Product** ↓	**Software Quality Assurance**	**Software Engineers**	**Testing**
User Interface Module	Rohit Wagh Krishna Soni	Jigar Shah Krishna Soni	Rohit Wagh Jigar Shah Krishna Soni
Authentication and Authorization module	Rohit Wagh Jigar Shah	Krishna Soni Rohit Wagh	Jigar Shah Rohit Wagh
Multipart File Uploading Servlet.	Rohit Wagh Krishna Soni	Krishna Soni Jigar Shah	Jigar Shah Rohit Wagh
File Uploading Components on Application Server	Jigar Shah Krishna Soni	Rohit Wagh Jigar Shah	Krishna Soni Jigar Shah

Table 8-1 Team Organization

8.3 SQA Purpose And Scope

- The SQA plan has under its scope all work products of this project, including source code, documents (Analysis, Design and Coding), Test Reports, etc.
- The SQA plan covers under it various documents, technical and others, which are created during the SDLC (as described in the documentation section ahead)

8.4 SQA Tasks And Responsibilities

8.4.1 Tasks:

Project Manager has defined the various tasks to be accomplished, sizing those tasks and grouping those tasks.

All have to finish their tasks with the planned quality, time and environment.

8.4.2 Responsibilities:

Project manager is responsible for familiarizing the co-workers with any project procedures facilities, and plans necessary to assure their effective integration into the project.

All are responsible for the successful completion of their module on time with the achievable and documented quality.

8.4.3 Documentation Section

This section describes the various documents produced during the software process and thus covered under the SQA activities, as listed below:

8.4.3.1 Project plan

In this document, we have estimated the amount of effort, cost and time required to complete the project. We have recognized the resources required, tasks involved and the division of the tasks among the members with clearly specified deadlines for each. By following the plan we aim to reach our deadlines in the time specified.

8.4.3.2 Risk Assessment and Mitigation Plan

Here, we have identified the various risks plaguing the project, assessing their probabilities, estimation their impact and establishing a contingency plan to prepare for the eventuality that the risk may turn into reality. Thus we assume a proactive stand in tackling potential problems, should they occur.

8.4.3.3 Software Configuration Plan

This document specifies the standard practices to be followed to control change. The major guidelines specify how to identify change, how to control change, how to ensure that change is being implemented, and how to report changes to all those concerned. We thus have a clear, unambiguous protocol to follow when the project is faced with major non-conformances with the plan or requirements.

8.4.3.4 Various system models like DFD, UML Diagrams

These models, which help depict the system in various perspectives, not only help understand the system better, but also help in communication the system functioning in a clear and unambiguous language, in effecting changes for analysis, in expanding the system and in innumerable other ways. These models will prove an invaluable tool in developing a high quality project in total conformance with the requirements and the design.

8.4.3.5 Software Test Plans

This document prescribes the various practices and procedures to be followed during the testing phase. It includes the various strategies, the test cases, the schedule of these tests and also the various levels at which the tests must be carried out. Thus, the testing activity is planned in advance, which helps us to make the testing process as close and relevant to the requirement as possible.

8.4.4 Minimum Work Products

The minimum necessary work products to be produced by the project are as mentioned in the Software Requirements Specification (SRS) in this document. As prescribed, all these work products should conform to the requirements set through use of maximum quality practices.

8.4.5 Standards, Practices And Conventions

During the project, we intend to follow various standard practices, which are expected of a quality software product. They are-

8.4.5.1 Coding Standards

We have laid out certain standard conventions that will be strictly adhered to during the project to ensure maximum readability, testability and maintainability of the code. These standards have been mentioned in the Coding Section of this document. They include aspects like naming conventions, commenting style, modularity, etc.

8.4.5.2 Reviews

We have laid out a standard protocol to be followed to review the various components prepared by us, suggest changes, and document the suggestions from various members towards improvement of the component or any other aspect of the software process.

8.4.5.3 Reporting

A protocol again has been laid out for reporting developments or changes to the other project members and the project guide. Strict adherence to these rules will ensure that all members and the project are always up-to-date with the project status.

8.4.5.4 Documentation

This will be included as a habit in every member for every kind of activity related to the project. This will enable us to track every major activity and action performed during the project that will facilitate easy tracing of mistakes and backtracking, if necessary.

8.4.5.5 Configuration Management

Various guidelines are laid out to dictate the everyday practices during coding and component development .How to handle source code during development, after partial completion, after full completion, hoe to backup source code, etc. are some of the issues decided upon here. They may be found in the SCM plan.

9. CONCLUSION

File uploading from remote computer is a critical element of enterprise security. While devising a secure solution, trade-off between performance and security should me maintained. Various other aspects like extensibility, scalability, reliability and cost of implementation should be taken into account. SecUp implement a very secure 3-tierd secure file uploading system, which is scalable, extensible and reliable. It provides user with many functionality including secure authentication and authorization, administration and logging activities. This project is completed successfully within given time period. Platform independent architecture makes SecUp deployable without recompiling code.

SecUp provides secure file uploading without increasing overhead of link encryption. It uploads file to internal network without storing it on the web server. It provides authentication and authorization at internal server making authentication information secured from eavesdropping.

10. REFERENCES

10.1 Books:

- Ed Roman, Rima Patel Shriganesh, Gerald Brose: Mastering Enterprise JavaBeans. 3rd Edition.
- James Goodwill: Developing Java Servlets.
- Herbert Schildt: The Complete Reference, Java 2, 5th Edition.
- Bruce Eckel: Thinking In Java.

10.2 Online References*:*

- www.ietf.org
- www.theserverside.com
- www.sun.java.com
- www.jguru.com
- www.ONjava.com
- www.oreilly.org

11. GLOSSARY

Web Server	The system which is the part of the Internet and lies in the public domain. It accepts http requests and sends back the http response or the html pages requested.
Domain Names	DOMAIN names are what people type into their browser when they want to visit your company's website.
Application Server	The system which is the part of the Intranet and lies in the private domain. It contains the actual business methods. It accepts IIOP requests and does the processing based on the requests. It then sends back the results after executing the business methods.
Servlets	Servlets are active server pages which lie in the web server and are used to perform operation in the web server and creating dynamic pages.
EJB	EJB stands for Enterprise Java Beans. It is a standard for building server side components in Java. It is actually a combination of Specifications & Set of java Interfaces. There are 2 types of beans. **Session Beans** and **Entity beans**.
Presentation Tier	This is the 1st tier of the 3-tier distributed architecture. It consists of the client browser and a link for communicating with the Web Tier.
Web Tier	This is the 2nd tier of the 3-tier distributed architecture. It consists of the web module/server and a link for communicating with the Presentation Tier and the Business Tier.
Business Tier	This is the 3rd tier of the 3-tier distributed architecture. It consists of the Application server having business methods deployed in it.

Acknowledgments

We would like to add a few cordial words for the people who were part of the project development in numerous ways and people who extended support & co-operation.

We fall short of words to thank our Head of Department **Prof. G.K.Kharate**, Project Coordinator and Internal Project Guide **Mr. J. A. Bharadwaj**, for all their expert guidance, encouragement and motivation during the development of this project, without which this project would never have been developed to its present form and complexity.

Our sincere thanks to our external project guides **Mr. Nirmal Juthani** and **Mr. Dhaval M. Shah** for their guidance and technical help. Without which development and completion of this project was impossible. We also acknowledge Internet community for their helping hands without any reimbursement.

We are very grateful to all our staff members of the K.K.W.I.E.E.R. for their magnificent support during our quest for more knowledge.

We owe a lot to our friends and families who helped us silently and encouraged us during the endless nights of toil.

- **Jigar Shah**
- **Krishna Soni**
- **Rohit Wagh**

Report Documentation & Accounting Page	
Report Code:	Report Number:

Address:
I.T. Department, K. K. Wagh Engineering College,
Hirabai Haridas Vidya Nagari, Amrutdham, Nashik
Pin – 422 003, M. S., INDIA
kkwcoe@bom6.vsnl.net.in

Report Title: "SecUp- Secure File Uploading System"

Authors:	**Authors Details:**
Jigar Shah(19) Krishna Soni (22) Rohit Wagh(24)	**Year** : 2004 – 2005 Branch: **Information Technology** Krishna Soni: krishnasoni2@rediffmail.com Rohit Wagh: rohit25_wagh@rediffmail.com Jigar Shah: jigar_a_shah83@yahoo.co.in

Type of Report: **FINAL**	**Time Covered From:** 2005/01/01 **To:** 2005/02/28 2 Months	**Date of Report:** 2005/ /	**Page Count:** 49

Keywords: Servlets, EJB, Application Server, Web Server, Domain Names.

Report Checked By:	Report Check Date:	Guide's Complete Name: Mr. J. A. Bharadwaj.	Total Copies: 5

Report Abstract:

Report Page (ii)