

**FILE MANAGEMENT SOFTWARE**

Low-Level Design Version Draft

|  |
| --- |
| **Team members** |

|  |  |
| --- | --- |
| **Employee id** | **Name** |
| 46291226 | SWETHA DEVISETTY |
| 46290147 | BHARGAVI DUMPA |
| 46290148 | ROHITHA ATTILI |
| 46290149 | DEVI SARANYA ADAPA |
| 46290150 | JYOSNA SRI KALABATTULA |

**Table of Contents**

**I) Low-Level Design**

| **1.Introduction** |  |
| --- | --- |
| 1.1 Purpose |  |
| 1.2 Document Conventions |  |
| 1.3 Intended Audience |  |
| 1.4 References |  |
| **2. Detailed system design** |  |
| 2.1 Design Description |  |
| 2.2 Flowchart |  |
| 2.3 Modules |  |

**Low-Level Design**

1. **Introduction**

The File Management software requirement specification provides an overview of the entire software. The entire SRS with overview description purpose, scope, tools used and basic description. The aim of this document is to gather, analyze and give an in-depth insight into the File Management Software by defining the problem statement in detail .The detailed requirements of the File Management Software are provided in this document.

**1.1 Purpose**

The purpose of this document is to show the requirements for the File Management Software which maintains the files in the hard disk and the client provides menu-based interface for user to interact and upload/ download /display /delete files to the file management app.

**1.2 Document Conventions**

TBD (To be continued).

**1.3 Intended Audience and Reading Suggestions**

This document is intended to be read by Client.

Intended Use:

* Development Team
* Maintenance Team
* Clients

**2. Detailed System Design**

**2.1 Design Descriptions:**

This project design mainly focuses on how to organize how files and folders are named , how nested folders arranges and how the files in the folders are handled by the clients.

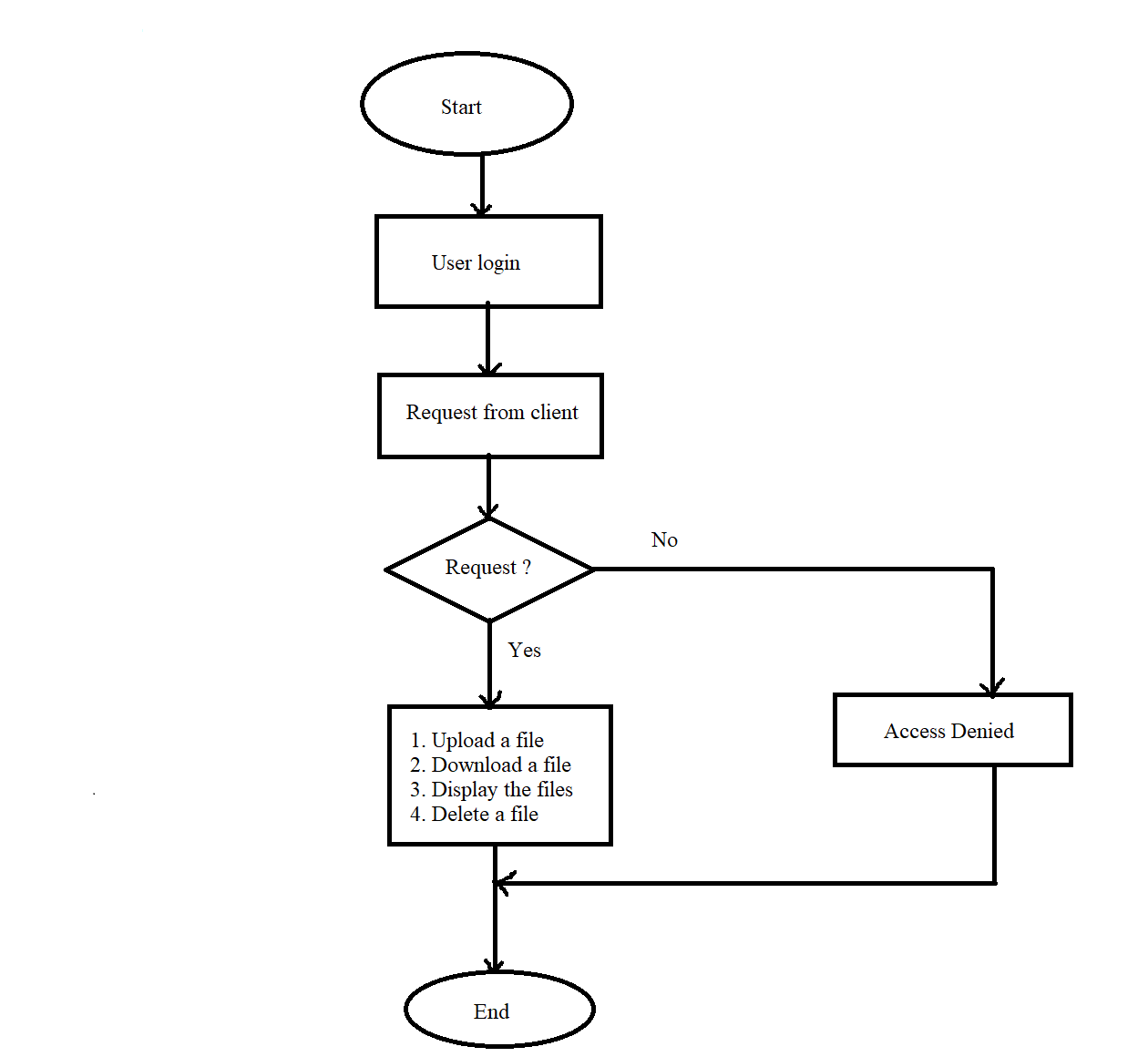
**Main menu:**

**Algorithm selection menu:**

**Algorithm sub menu:**

* The client is used to send a request to the server.
* After the server accepts the request that was send by client the client can access all the permissions allowed by servers.
* If the client want to download a file they can download.
* They have another permissions like Upload/Download/display/Delete.

**Flow Chart:**



**2.3 Modules** :

**2.3.1 client .c**

**main()** :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | main | | | |
| **Input** | Parameter Name | NA | Initial value | NA |
| **Output** | Return value type | NA | -  Text Box | - |
| **Description** | The main function will display options to the user | | | |
| **Pseudo**  **Code** | 1.Login  2.Logout  3.Enter the choice | | | |

**2.3.2 login.c()** :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | login.c | | | |
| **Input** | Parameter Name | char | Initial ValueText Box | NA |
| **Output** | Return value type | int | Text Box- | - |
| **Description** | The authenticated user who has access to the server should login with the username and password. | | | |
| **Pseudo Code** | 1.Username  2.Password | | | |

**2.3.3**  List Download\_file():

| **Name** | Download File | | | |
| --- | --- | --- | --- | --- |
| **Input** | Parameter Name | char | Initial Value | - |
| **Output** | Return Value type | int | - | - |
| **Description** | The user can see all the downloaded files which are present in the server. | | | |
| **Pseudo**  **Code** | 1.All Downloaded Files.  2.Data Received.  3.No Data Found. | | | |

**2.3.4 List\_all files():**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | List all files | | | |
| **Input** | Parameter Name | char | Initial Value | - |
| **Output** | Return Value type | int | -  Text Box | - |
| **Description** | It displays the user drive files | | | |

**2.3.5 Download\_file()**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | Download file | | | |
| **Input** | Parameter Name | char | Initial Value | - |
| **Output** | Return Value Type | int | -  Text Box | - |
| **Description** | The user want to download a particular file He can download it. | | | |
| **Pseudo**  **Code** | 1.Enter File Name  2.Downloaded contents | | | |

**2.3.6 Upload\_File():**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | Upload File | | | |
| **Input** | Parameter Name | char | Initial Value | - |
| **Output** | Return Value Type | int | - | - |
| **Description** | The user wants to upload a file to the server He can. | | | |
| **Pseudo**  **Code** | 1.Please enter the file name to upload.  2.File opened failed(if the file is not uploaded successfully).  3.File open successfully.  4.Downloaded content. | | | |

**2.3.7 Delete\_file ():**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | Delete file | | | |
| **Input** | Parameter Name | char | Initial value | - |
| **Output** | Return value type | int | Text Box | - |
| **Description** | If the user wants to delete a particular file the user can. | | | |
| **Pseudo**  **Code** | 1.Please enter a file name to be deleted.  2.File deleted Successfully.  3.Successfully logout. | | | |