Table of Contents

[Project Background 3](#_Toc350710321)

[Problem Statement 3](#_Toc350710322)

[Project Vision 3](#_Toc350710323)

[Actors 5](#_Toc350710324)

[Assumptions 5](#_Toc350710325)

[Project Requirements 5](#_Toc350710326)

[Requirement A – Subscription Plans 5](#_Toc350710327)

[Requirement A1 – Plan Duration 5](#_Toc350710328)

[Requirement A2 – Plan Expiry 6](#_Toc350710329)

[Requirement A3 – Purchase of New Plan 6](#_Toc350710330)

[Requirement A4 – Renewal of Subscription Plan 9](#_Toc350710331)

[Requirement A5 – Cancellation of Subscription Plan 12](#_Toc350710332)

[Requirement B – Verification System (Deals with Verification of User Credentials) 14](#_Toc350710333)

[Requirement B1 – Verification of Login Credentials 14](#_Toc350710334)

[Requirement B2 – Verification of Subscription Validity 14](#_Toc350710335)

[Requirement B3– Verification of Upload Quota 14](#_Toc350710336)

[Requirement C – Backup & Restore System (Automatic Data Backup) 16](#_Toc350710337)

[Requirement C1 – Scheduling of Automatic Data Backup Process 16](#_Toc350710338)

[Requirement C2 – Specifying Data to Backup 16](#_Toc350710339)

[Requirement C3 – Notification of Automatic Backup 16](#_Toc350710340)

[Requirement C4 – Identification of each Automatic Data Backup Session 16](#_Toc350710341)

[Requirement C5 – Restoring Automatically Backed-up Data 16](#_Toc350710342)

[Requirement D –Backup & Restore System (Manual Data Backup) 18](#_Toc350710343)

[Requirement D1 – Specifying Data to Backup 18](#_Toc350710344)

[Requirement D2 – Restoring Manually Backed-up Data 18](#_Toc350710345)

[Requirement E – Data Transfer Process Engine (Used by Manual & Automatic Backup Systems) 20](#_Toc350710346)

[Requirement F – Security Requirements 22](#_Toc350710347)

[Requirement F1 – Password encryption 22](#_Toc350710348)

[Requirement F2 – Secure transmission of data to/from FTP Server 22](#_Toc350710349)

[Requirement G – Application Start-up Requirements 22](#_Toc350710350)

[Key Application Components 22](#_Toc350710351)

[Requirement H – Key Application Components 22](#_Toc350710352)

[Requirement H1 – FileBird Component 22](#_Toc350710353)

[Requirement H2 – UI Component 22](#_Toc350710354)

[Requirement H3 – FTPS Client Component 23](#_Toc350710355)

[Requirement H4 – FTP Client Component 23](#_Toc350710356)

[Requirement H5 – Verification System Component 23](#_Toc350710357)

[Requirement H6 – Automatic Backup System 23](#_Toc350710358)

[Requirement H7 – Manual Backup System 24](#_Toc350710359)

[Requirements H8 – Data Transfer Engine 24](#_Toc350710360)

[Requirement H9 – Logging System 24](#_Toc350710361)

[1) One server will store all the user data along with their ip and userid and password. The same details would be replicated on another mirror server. The user will first be taken to the first server if that is working it will tan take him to the respective ftp server and if that server is down it will take him to the mirror for verification. --------NOTED 26](#_Toc350710362)

[High Level Interaction Diagram 26](#_Toc350710363)

# Project Background

Jeet Technologies provides FTP backup solution to its users on a subscription basis. There are several subscription plans, such as 2GB, 3GB etc. And each plan is valid for 365 days, upon which user is required to renew his plan.The company owns about 10 FTP srvers and a user is allocated space to one of these servers by the administrator.

# Problem Statement

Current setup makes use of Filezilla FTP client to allow users to backup their data to FTP server.

The 2 major problems with this setup are

1. Filezilla lacks support for automatic data backup
2. There is no automated way of reminding the user, when his subscription plan is about to expire.

# Project Vision

To build a FTP solution, which

1. Allows users to backup data (automatically, as well as manually) to a FTP server.
2. Notifies the user when his subscription is about to expire. Notification is served on 30th day, 20th day, 15th day, 10th day, 5th day & on the last day.

# Actors

|  |  |
| --- | --- |
| **Actor** | **Role in the system** |
| User | * Backs up data * Restores data * Sets automatic backup intervals (daily/weekly/monthly) * Selects the files for automatic backup |
| Administrator | * Allocating storage space for the user on the FTP server   HOW ? Can you explain in detail  FTP Server provides the functionality of allocating storage quota for each user. This has to be done manually by the Administrator   * After the user renews his subscription plan after expiry, the Administrator manually re-activates user’s FTP account.   HOW ? Can you explain in detail  FTP server also provides feature to enable/disable user account. So, Administrator has to manually enable/disable user account depending on his subscription plan.. |

# Assumptions

1. Current setup allows the users to buy, renew, cancel subscription plans through the website, www.estorage.in

# Project Requirements

## Requirement A – Subscription Plans

### Requirement A1 – Plan Duration

Subscription plan’s duration must be either 1 year, 2 years, 3 years, 4 years or 5 years . The subscription plan starts from first day of data upload.

The subscription could be of 1year, 2 years, 3 years, 4 years and a 5 years.

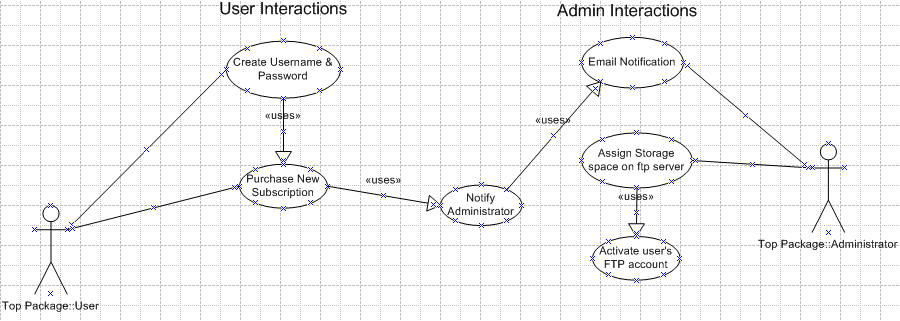
### Requirement A2 – Plan Expiry

User must be notified when his subscription is about to expire. Notification must be served on 30thday, 20thday, 15thday, 10thday, 5thday & on the last day.

### Requirement A3 – Purchase of New Plan

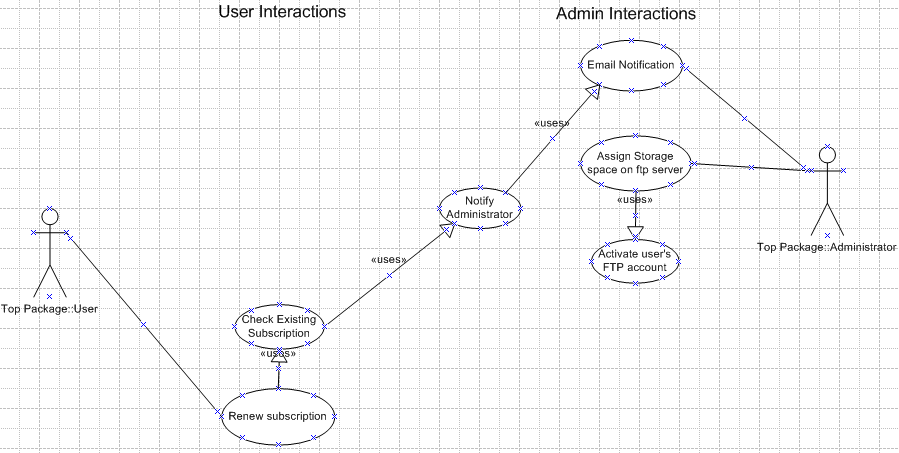
User should be able to buy a new subscription plan via the website.

HOW ? Can you explain in detail – I am assuming here that you currently have a website, through which user can buy, cancel and renew subscription plans



|  |  |
| --- | --- |
| **User buys a new subscription plan** | |
| Preconditions | None |
| Primary Actors | User |
| Secondary Actor | Administrator |
| Interaction Flow | |  |  |  | | --- | --- | --- | | Step | User | Administrator | | 1. | User visits the website |  | | 2. | User registers |  | | 3. | User buys a new subscription plan  He should get a popup that the plan would start in 24 hrs  Again, this refers to your current website. | Email notification is sent to Administrator | | 4. |  | Administrator assigns storage space on the FTP server and activates user’s ftp account | | 5. | Email notification is sent to the user |  | |

### Requirement A4 – Renewal of Subscription Plan

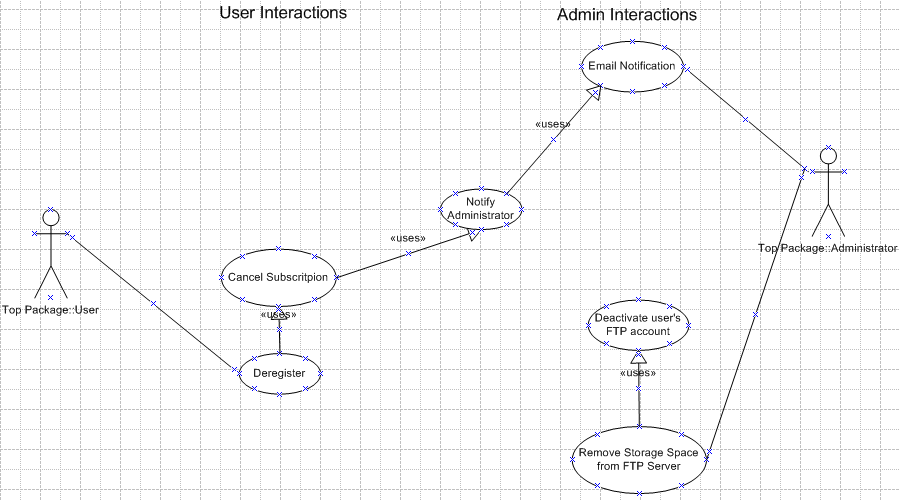
User renews subscription plan

|  |  |
| --- | --- |
| **User renews subscription plan** | |
| Preconditions | * User must have an active/expired plan * User can buy a new plan even if his current plan has not expied. |
| Primary Actors | User |
| Secondary Actor | Administrator |
| Interaction Flow | |  |  |  | | --- | --- | --- | | Step | User | Administrator | | 1. | User visits the website |  | | 2. | User upgrades/downgrades/renews previous subscription plan | Email notification is sent to Administrator | | 3. |  | Administrator upgrades/downgrades storage space on the FTP server | | 4. | Email notification is sent to the user |  | |

How does ftp server storage is assign ? i assume it is mannaul the way it is done.

Yes, user quota has to be assigned manually on ftp server.

### Requirement A5 – Cancellation of Subscription Plan

User cancels subscription

|  |  |
| --- | --- |
| **User cancels subscription** | |
| Preconditions | * User must already have an account. * It is not mandatory for user to own a subscription plan. |
| Primary Actors | User |
| Secondary Actor | Administrator |
| Interaction Flow | |  |  |  | | --- | --- | --- | | Step | User | Administrator | | 1. | User visits the website |  | | 2. | User de-registers | User’s username & password are automatically removed from the system | | 3. |  | Email notification is sent to Administrator | | 4. |  | Administrator manually deactivates user’s ftp account. | | 5. | Email notification sent to user |  | |
|  |  |

## Requirement B – Verification System (Deals with Verification of User Credentials)

### Requirement B1 – Verification of Login Credentials

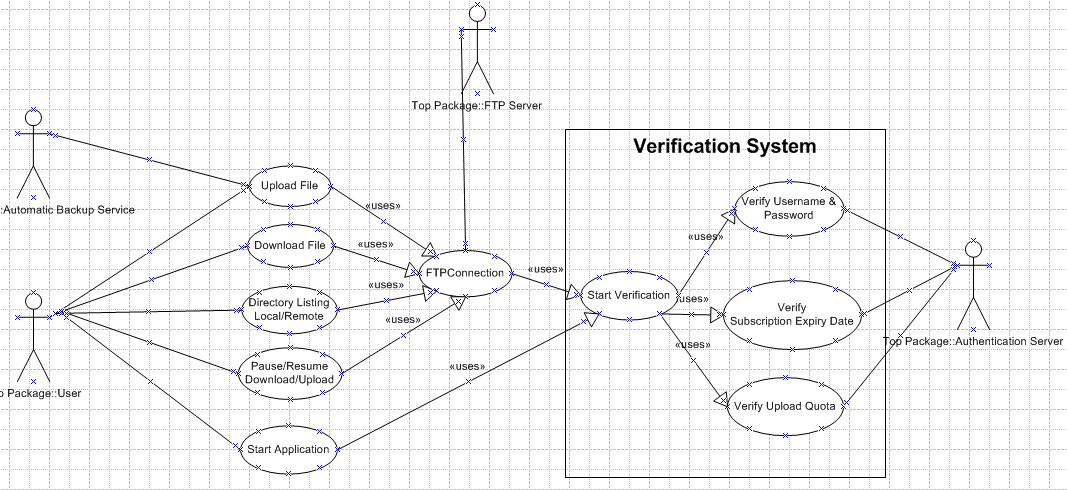
1. Username and password will be verified at application start-up and at the time of establishing FTP connection
2. If username and/or password is incorrect, user will be notified with the message “Invalid username and/or password”
3. On 3 incorrect attempt the user should be locked and only after the admin unlocks him it should start working. A message should pop up stating user locked contact admin on info@estorage.in

### Requirement B2 – Verification of Subscription Validity

1. Subscription expiry date will be verified at application start-up and at the time of establishing FTP connection
2. If user’s subscription has expired, user will be notified with this message “Your subscription has expired. Please renew your subscription to continue backing up your files”

### Requirement B3– Verification of Upload Quota

1. Upload quota will be verified at application start-up and at the time of establishing FTP connection
2. If user has finished his upload quota, then he will be notified with this message “You have reached the maximum upload quota, as per your subscription plan. Please renew your subscription to continue backing up your files”
3. An email of user overquota should be send to the admin. [info@estorage.in](mailto:info@estorage.in)



## Requirement C – Backup & Restore System (Automatic Data Backup)

Application must support automatic backup of data.

### Requirement C1 – Scheduling of Automatic Data Backup Process

1. User should be able to configure the application to backup data on a daily / weekly / monthly basis
2. User should also have the option to specify the time, at which the automatic backup process should start
3. Incremental or Overwrite option. If the user has 500gb of database which he wants incremental how would we handle that. Can we have a incremental option where he can just upload the data which has been modified. -----------I will look into how this can be done.

### Requirement C2 – Specifying Data to Backup

1. User should be able to specify files & folders to automatically backup

### Requirement C3 – Notification of Automatic Backup

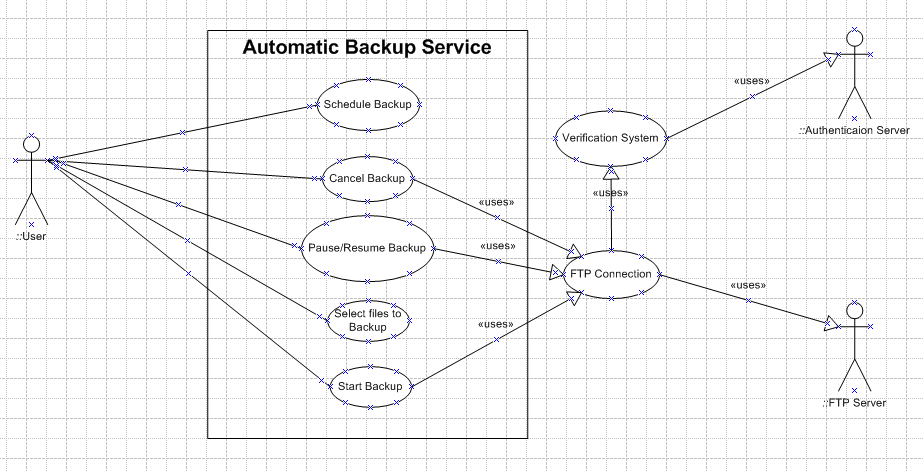
1. User should be notified via task bar notification popup, when the automatic backup process starts

### Requirement C4 – Identification of each Automatic Data Backup Session

1. By default, the application must name each backup session in dd/mm/yy format.
2. Application must maintain a list of all files associated with the backup session in the database
3. Optionally, user should be presented with the option to add some comments on each backup

### Requirement C5 – Restoring Automatically Backed-up Data

1. When user wants to restore a backup session, all files related to selected backup session must be downloaded from the server onto local hard drive.



## Requirement D –Backup & Restore System (Manual Data Backup)

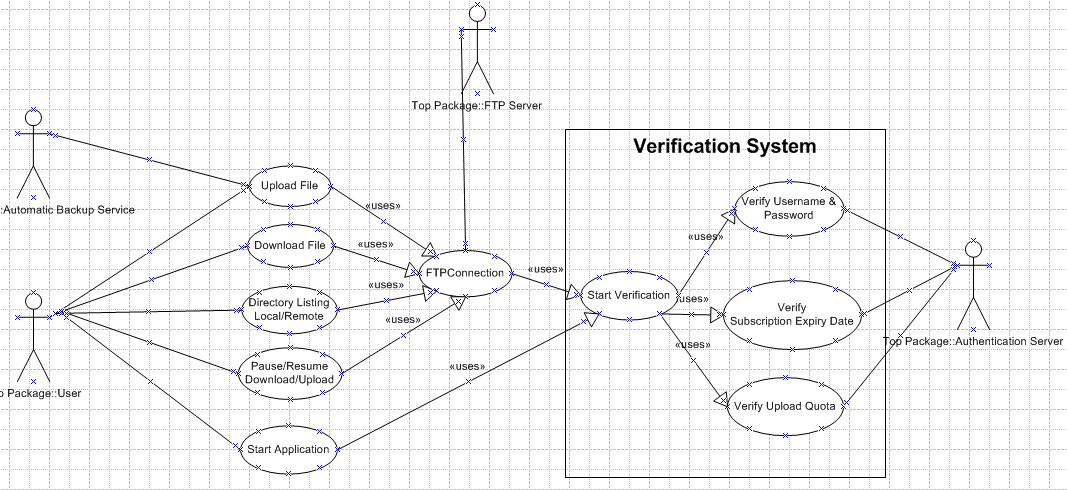
Application must allow user to manually backup data

### Requirement D1 – Specifying Data to Backup

User should be able to manually select multiple files and folders he wants to backup.

### Requirement D2 – Restoring Manually Backed-up Data

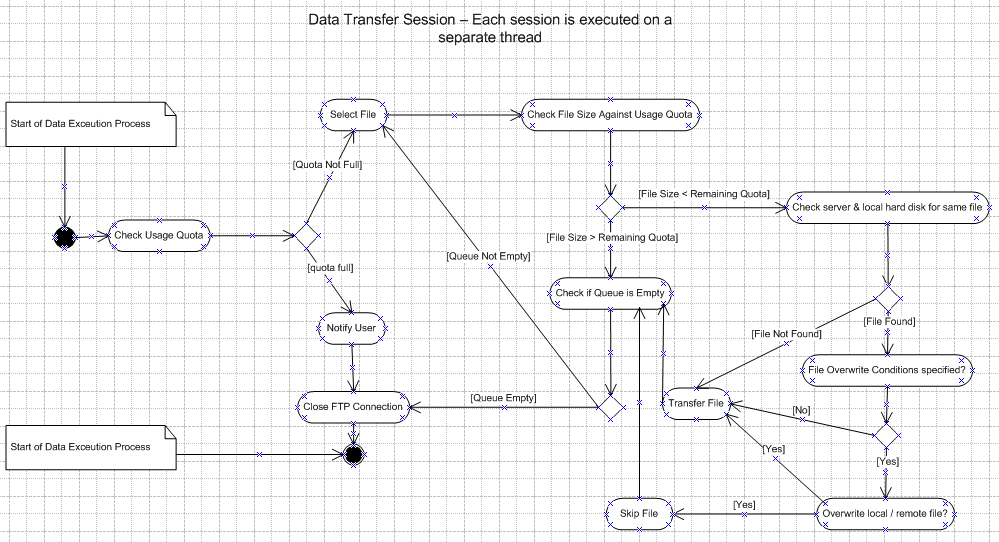
User must be able to manually select the files he wants to restore



## Requirement E – Data Transfer Process Engine (Used by Manual & Automatic Backup Systems)

Data transfer refers to upload / download of data. Data transfer process requirements are applicable to manual as well as automatic data backup as well as data restore.

1. User should be able to pause / resume / cancel the process
2. User must be able to select target folder for upload / download
3. During execution of the process, application must display
   1. List of all files that were successfully backed-up / restored.
   2. List of all files that could not be backed up / restored, along with appropriate explanation / error code for failure.
   3. Current file that is being backed up / restored, along with percentage of the file uploaded / downloaded.
4. Before backing –up each file, remaining storage space for the user must be determined.
   1. If the storage space on FTP server is not zero, select a file for upload
      1. If there is insufficient storage space for the selected file, skip that file and select the next file in queue. But a message should be poped for the skipped file. ----------A list of all skipped files will be displayed at the end of backup session. A pop-up message is very irritating for the user. Or we can display a notification in notification bar.
5. The user should also be presented with below options
   1. Overwrite local / remote file.
   2. Overwrite local / remote file only if file on the server & local hard disk are the same versions of the file.
   3. Overwrite local file only if file on the server is an older version and file on the local hard disk is a newer version.
   4. File versions must be determined by comparing file creation and last modification timestamps of both local and remote files. The comparison must be done before starting the download / upload of the file
   5. Incremental Backup: As asked before this is an important aspect.



## Requirement F – Security Requirements

### Requirement F1 – Password encryption

1. User’s password will be encrypted using MD5 encryption algorithm
2. The encrypted password will be stored in a configuration file.

### Requirement F2 – Secure transmission of data to/from FTP Server

1. Application must use FTPS (FTP over SSL), to ensure all communication with the server happens in a secure, encrypted form.
2. In case the FTP server does not support FTPS, application must gracefully fall back to traditional FTP

## Requirement G – Application Start-up Requirements

* User must be given the option to automatically launch the application during OS boot
* During start-up, application will read all configuration data from the configuration file.

# Key Application Components

## Requirement H – Key Application Components

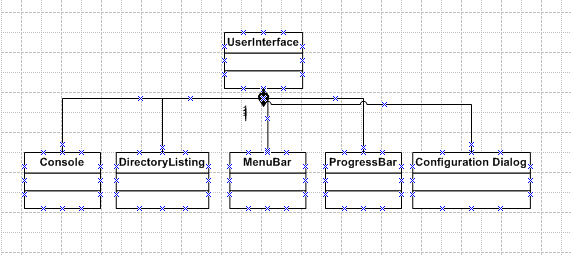
Based on the above requirements, the application must be broken down into following key components

### Requirement H1 – FileBird Component

1. The FileBird component is the entry-point into the application
2. It initializes all other components.
3. FileBird component must run on the main thread.
4. UI Component, FTPs Client Component, FTP Client Component must all run on separate threads

### Requirement H2 – UI Component

This component must provide a set of relevant user-interface components for the user to manage his data



### Requirement H3 – FTPS Client Component

1. This component is invoked by the User Interface component
2. Initialization parameters such as sever IP, username, password are supplied to this component by the UI component
3. This component should communicate with the User Verification Component & the FTP Server
4. FileBird component can create multiple instances of this component
   1. Each instance of this component must run a separate thread

### Requirement H4 – FTP Client Component

1. In case FTP server does not support FTPS, then FileBird component must automatically fallback to the FTP Client component.
2. All functional requirements for this component must be the same as that of FTPS Client component

### Requirement H5 – Verification System Component

Please refer to Requirement B for more details

### Requirement H6 – Automatic Backup System

Please refer to Requirements C,D & E

### Requirement H7 – Manual Backup System

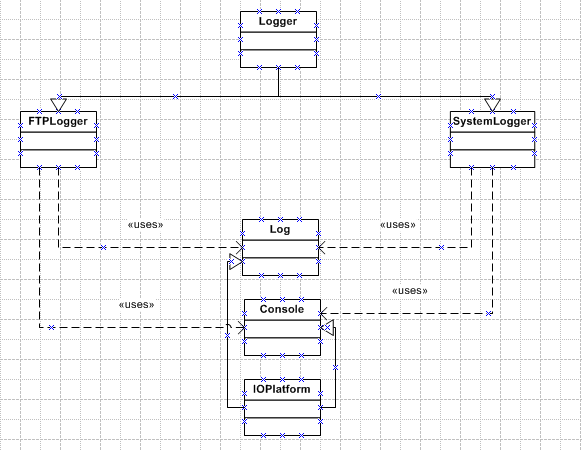
Please refer to requirements C, D & E

### Requirements H8 – Data Transfer Engine

Please refer to requirement C,D & E

### Requirement H9 – Logging System

* All logs must be logged to .log files
* Logging system must be composed of FTP Logger & System Logger
* FTP logger must:
  + Create a separate log file for every backup & restore session
  + Name the log file with name of backup / restore session
  + Log start and end time of the session, along with files that were successfully backed-up / restored and files which could not be backed-up / restored (along with the error codes)
* System logger must
  + Log all system-critical errors
  + Name the log file in dd / mm / yy format



# One server will store all the user data along with their ip and userid and password. The same details would be replicated on another mirror server. The user will first be taken to the first server if that is working it will tan take him to the respective ftp server and if that server is down it will take him to the mirror for verification. --------NOTED

# High Level Interaction Diagram

