FogWire

Software Requirements Specification

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Date | 10/27/2009 | Authors | Anthony Sinatra  Dmitry Sharlot  Joseph Everett  Ricardo Viera | Version | 1.0 |
|  | 10/27/2009 |  | Anthony M Sinatra | Created specification overview | |

Table of Contents

PROJECT DRIVERS

1. The Purpose of the Product

2. Client, Customer and other Stakeholders

3. Users of the Product

PROJECT CONSTRAINTS

4. Mandated Constraints

5. Naming Conventions and Definitions

6. Relevant Facts and Assumptions

FUNCTIONAL REQUIREMENTS

7. The Scope of the Product

8. Functional and Data Requirements

NON-FUNCTIONAL REQUIREMENTS

9. Look and Feel Requirements

10. Usability Requirements

11. Performance Requirements

12. Operational Requirements

13. Maintainability and Portability Requirements

14. Security Requirements

**1.)Purpose of the Project**

Many people communicate with one another using electronic mail. Most people believe that their communication remains secure; however, this is not the case. These electronic mails travel over many networks in order to reach the intended recipients. These networks may exhibit security flaws and various vulnerability points, which could compromise the user’s data.

Communications and/or data which has been intercepted during transit should remain restricted from use by a third party.The FogWire application will enable the user to secure his/her electronic mails prior to sending. Only verified users will be able to unlock these electronic mails upon receipt.

**2.) Client, Customer and Other Stakeholders**

2.1. Our client is the general public.

2.2. Our customer is any person who sends electronic mail and would like to ensure that it is sent securely.

2.3. Other stakeholders

Pace University and Professor Marchese have provided the initiative for the development of this product.

**3.) Users of the Product -** Anthony

4. Mandated Constraints - Anthony

5. Naming Conventions and Definitions - Anthony

6. Relevant Facts and Assumptions - Anthony

7.)Scope of the Product

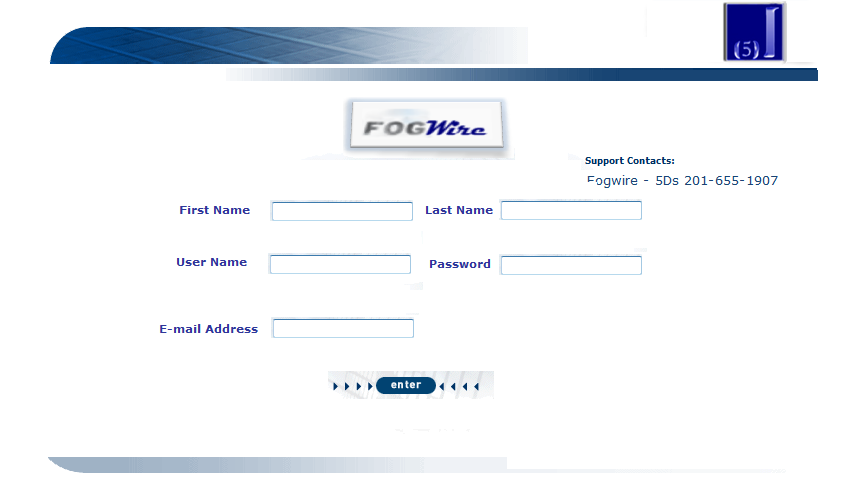
* + Users require the ability to setup a user profile within the application.
  + The user profile will authenticate each user in order to verify whether he/she should be given access to lock and/or unlock data communication.
  + Users require the ability to securely communicate with one another.
  + Users need to securely transfer data from one authorized user to another while restricting unauthorized access.
  + If data interception occurs, the data should remain unusable by another party.
  + Users need to obtain and utilize protected data which has been sent from another authorized user
  + Users will send this protected data via an electronic mail client.

8.)Proposed Functionality

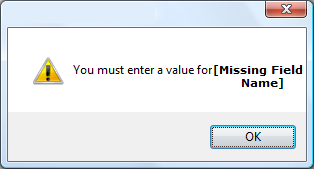
8.1) User profile setup

When users launch the application for the first time, he/she will have the ability to setup a user profile. This profile will store information about the user:

1. First Name
2. Last Name
3. User Name
4. Password
5. E-mail Address
6. Unique Private Key (**Note:** This data point will not be displayed to the user. It will be stored in the database)



Note: All Field are required. User cannot save a profile if any of the fields are incomplete. If any of the required fields have been omitted, the user will be receive a warning:

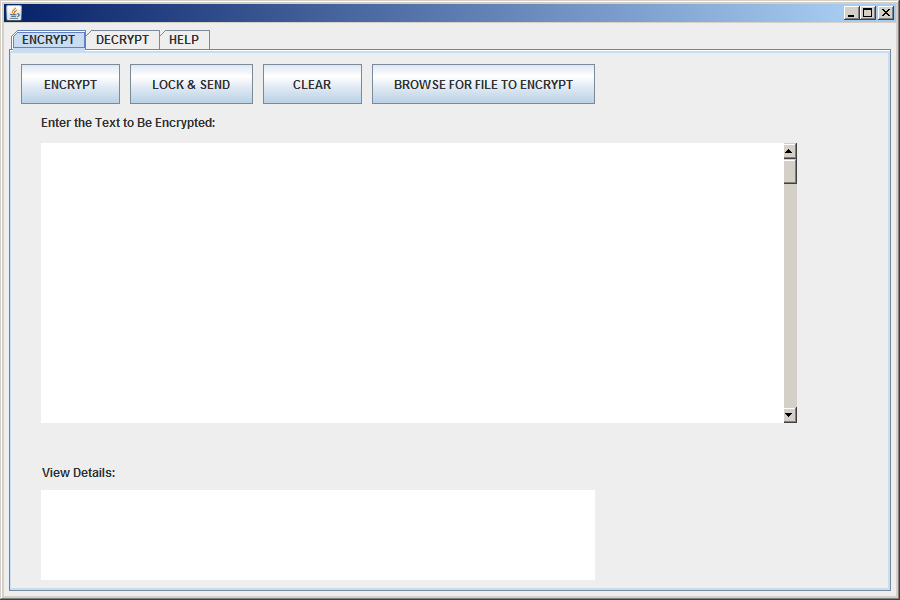


After the data has been entered, the application will connect to the FogWire database server in order to store the information which has been entered. The user’s private key will be generated by the system and stored in the database with the user’s information.

**Note:** The users’ private key will not be communicated to the user.

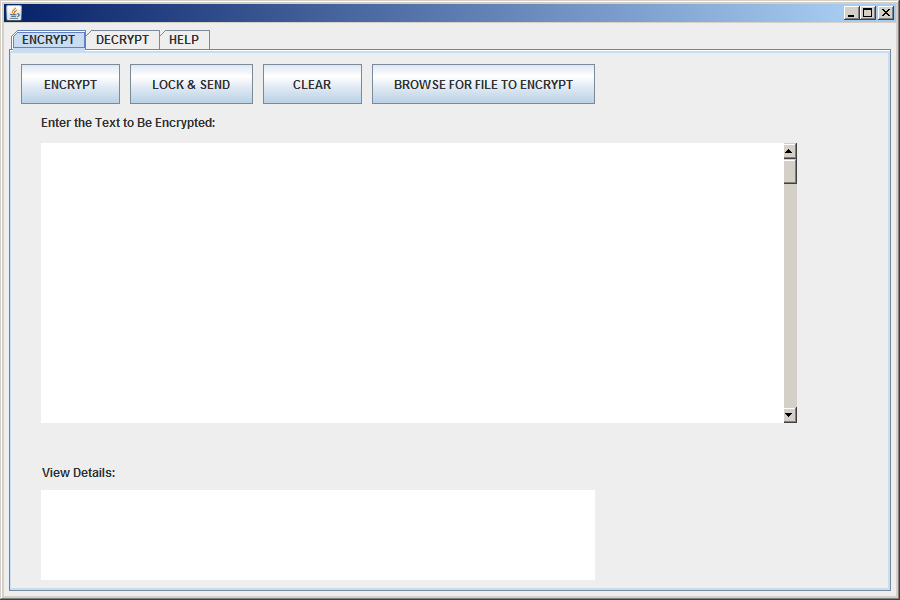
8.2) Securing Data

After the user has setup his/her profile information, he/she will access the application. In this area, the user will be able to free-type data into the text box or select a file to secure.



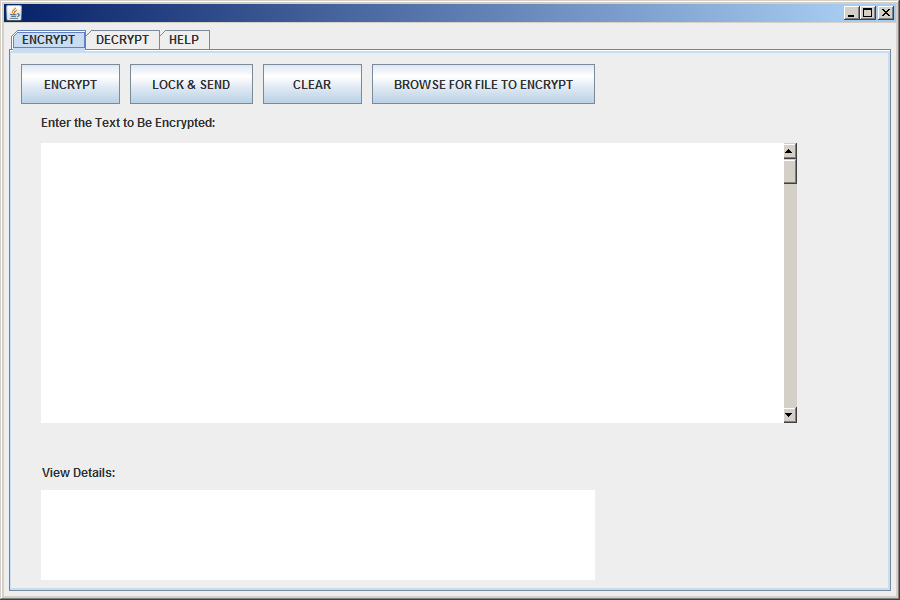
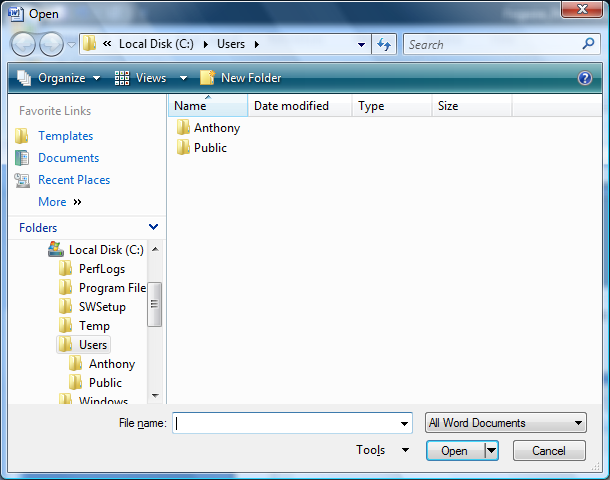
a.)The encrypt tab will be selected by default after the user has opened the application and verified his/her login information.

b.) If the user selects to type data into the ‘Enter the Text to be encrypted’ area, the ability to browse for a File to encrypt will be disabled.



User can free-type text and/or cut and paste text from another word processing application into the text box.

c.)If the user selects to browse for a file, a prompt will appear which allows the user to select a file from using a standard file selector:



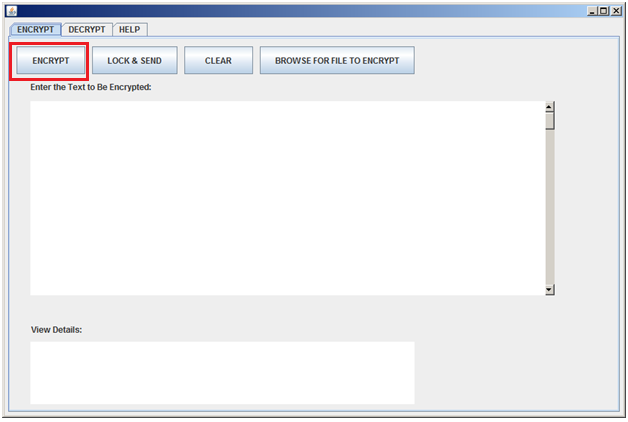
The file selector will default to the users’ C:/ drive.

d.) After the user has selected a file or entered information into the text-box, he/she can select to:

1. Encrypt
2. Lock and Send
3. Clear

8.3) Encryption

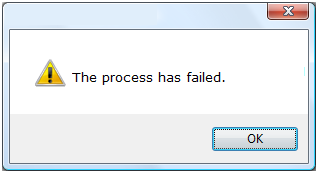
Data entered into the textbox or the file selected will become encrypted after the using selects the ‘Encrypt’. The encryption process will transform the data or data file into an encrypted file which would be completely undecipherable. The encryption process will insert the private key of the user into the data at each encryption point. This will allow the ‘Decryption’ process to unlock the data. **(Decryption process will be described in section 2.5)** The encrypted file will be stored on the users’ computer in a default location which the user can select in a setup area.



After the encryption process has been successfully completed, the user will receive a message which informs he/she that the process was completed successfully.

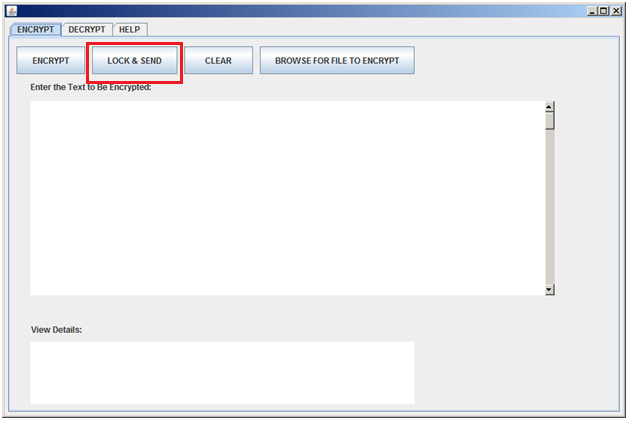
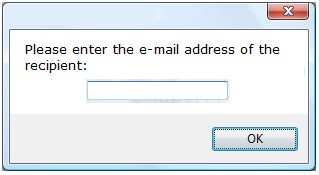


If there is a problem with the encryption process, the user will receive a message which states the process has failed.



8.4) Lock and Send

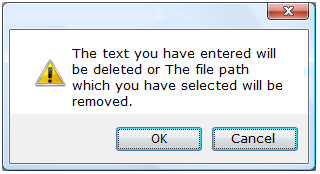
Lock and Send will follow the same process described above; however, there will be an additional step. The user will send the encrypted file to another FogWire user. After the encryption process has been completed, the user will receive a prompt which asks him/her to enter the e-mail address of the recipient.



The encrypted file will then be sent to the recipient via the user’s e-mail client using the from e-mail address which has been stored for the user of the application. The e-mail body will contain a message “You have received a message from another FogWire user. Please open the file using the FogWire application in order to unlock the data”. The encrypted file will be an attachment to the e-mail. It will be labeled with the file extension ‘.FOG’.

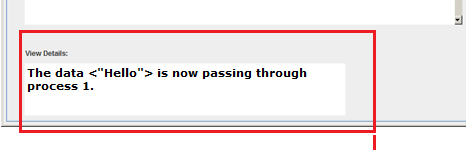
8.5) Clear

When the user selects ‘Clear’, the text box will be emptied of all the data which has been entered. If the user has selected a file, the file path will be cleared. The user will receive a warning message:



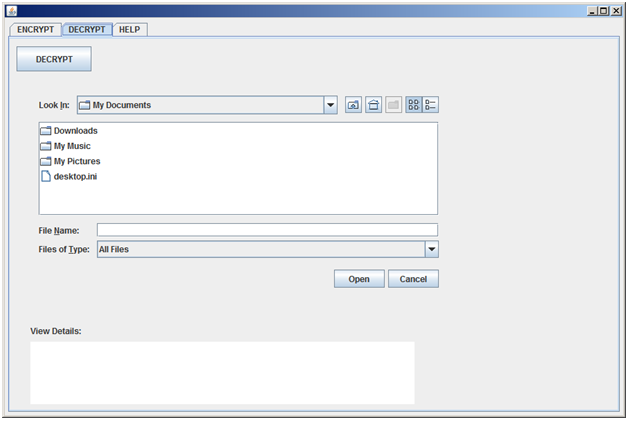
8.7) View Details

The user can select to view details which will display the encryption process to the user on the interface. The detailed steps of the encryption process will be explained to the user:

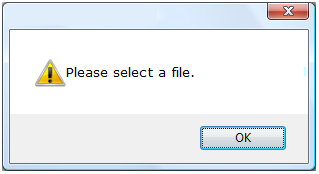


8.8) Decryption

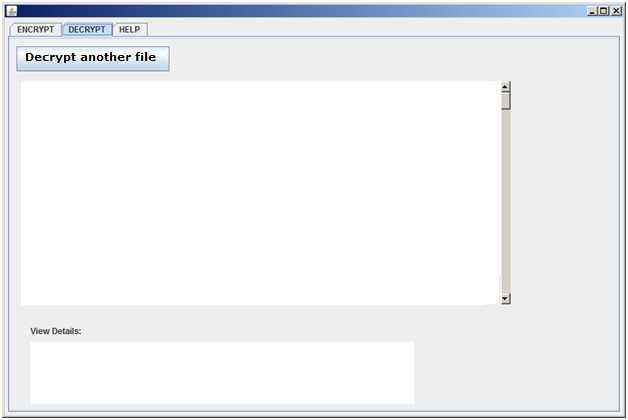
In order for the encrypted data to be unlocked, the recipient of the encrypted file **must** open using the ‘Decrypt’ area of the FogWire application.



The user can only select file types with ‘.FOG’ extension types, since these are the files which FogWire has previously encrypted. If the user does not select a file and he/she selects ‘Decrypt’, a prompt will appear:



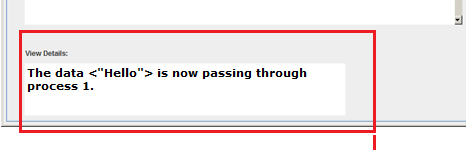
Once the file has been decrypted, the text will display to the user:



The user can also select to ‘Decrypt another file’ which will bring the user back to the file selector screen.

8.9) View Details (Decryption)

The user can select to view details which will display the decryption process to the user on the interface. The detailed steps of the encryption process will be explained to the user:



8.10) Implications

The application will not function properly unless the user is connected the internet. The application **must** have a connection to the database in order for the system to verify and/or store the users’ information.

8.11) Database

A table to store:

1. First Name
2. Last Name
3. User Name
4. Password
5. E-mail Address
6. Unique Private Key

9. Look and Feel Requirements - Joe

10. Usability Requirements- Joe

11. Performance Requirements - Ricardo

12. Operational Requirements – Dmitry

13. Maintainability and Portability Requirements- Ricardo

14. Security Requirements- Dmitry