

## Team 3 Design Document

For our project we decided to use a combination of the information expert, high cohesion and low coupling methodologies. The Main activity of our project is a file explorer that delegates certain tasks to other classes while also implementing a few of the use cases itself. By having the Main activity do all of the smaller use cases we can maintain a high cohesion for all of our other classes as well as promoting a lower coupling. Both the high cohesion and information expert design patterns help support the low coupling design pattern. Having high cohesion in many classes that are called only by the information expert greatly reduces the amount of coupling in the project.

The Delete File, Search Copy then Paste File and Copy then Paste file ALT flow System Diagrams all show that they are handled by the Main activity in order to keep the other classes more focused on their own responsibilities. Each of these use cases are also only important for the Main activity to know about which supports both the high cohesion and information expert design patterns.

The Store File System Diagram shows that we use another class called store to store the data onto a server using FTP. This was done to enforce the High cohesion methodology. The Store class is only called by the Main activity which supports the low coupling design methodology.

The Class diagram shows that after the user logs in the Main Activity takes over and delegates a lot of the extra responsibilities of the application into more focused and specialized classes, such as Encrypt and Store. The setup page, start page, change password and login page all have their own separate responsibilities and none of them interact with each other, which follows the low coupling design pattern.