

Software Specification

Software Req. Specification Format.

Classical Method:

1. Product overview and summary
2. Development, Operation and Maintenance Environment
3. External Interface and Data flow, High level data flow diagram, display report formats
4. Functional Requirements
5. Performance Requirements
6. Exception Handling
7. Foreseeable Modifications and Enhancement
8. Acceptance Criteria

Object Oriented Method:

1. Product overview and summary
2. Development, Operations and Maintenance Environment
3. User-case Diagram and description
4. Class diagram including collaboration and sequence diagram
5. performance requirements
6. exception handling
7. foreseeable modification and enhancement
8. acceptance criteria

1. Product overview and Summary

Product will provide a command line interface that allows the user to embed a document that is either plain text, binary or encrypted inside of an image file. The cli interface will support jpeg, gif and png file format. It will embed append the data to the end of the image file in such a way as to not corrupt the image file. The user data will be denoted by a unique eof marker. A log file will be created in the current working directory (the directory where the program is executed from) and record all operations performed. The goal of the program is to make the data as transparent as possible, making the file seem like it is untouched.

2. Development, Operation and Maintenance Environment

Development will be done exclusively in C++, the operation environment will be Linux/Unix file system. Program may be compiled in other environment if the proper libraries are present, but no support will be provided. No maintenance should be required, the log file may need to be flushed out occasionally. User support will be provided for 10 a year block at customer request.

3. External Interfaces and Data flow.

See attached insert

4. Functional Requirements

A file to be embedded, an image file that's supported by the application (JPEG, GIF, PNG). A Linux/Unix environment is preferred.

5. Performance Requirements

Minimum requirements would be any modern computer running a Linux or Unix environment with all the proper library required to compile the program. Theoretically,

6. Exception Handling

Exception Handling should deal with omitted filenames, zero sized files, incorrect paths, among any other problems that may occur while developing the product. The program will deal with all foreseeable user errors and

7. Foreseeable Modification and Enhancements

Graphical front end to be added for more a more use friendly approach. Support for more image formats. Extend the application to support other media (sound, video, etc). Add encryption for data being stored.

8. Acceptance Criteria

If image is embedded successfully and if the image can be extracted successfully.