Software Requirements Specification

for

Image Stenography with

AES Encryption

Version 6.0.0

Prepared by:

Nathan Beyer

Chris Hoegger

Chris Menning

Leilani Ray

Legendary Lichens

November 17, 2017

Table of Contents

[Table of Contents i](#_Toc497414610)

[Revision History ii](#_Toc497414611)

[1. Introduction 1](#_Toc497414612)

[1.1 Purpose 1](#_Toc497414613)

[1.2 References 1](#_Toc497414614)

[2. Overall Description 1](#_Toc497414615)

[2.1 User Classes and Characteristics 1](#_Toc497414616)

[2.2 Operating Environment 1](#_Toc497414617)

[2.3 Design and Implementation Constraints 1](#_Toc497414618)

[2.4 Assumptions and Dependencies 1](#_Toc497414619)

[3. External Interface Requirements 2](#_Toc497414620)

[3.1 User Interfaces 2](#_Toc497414621)

[3.2 Hardware Interfaces 2](#_Toc497414622)

[3.3 Software Interfaces 2](#_Toc497414623)

[3.4 Communications Interfaces 2](#_Toc497414624)

[4. System Use Cases 3](#_Toc497414625)

[4.1D Overall Use Case – Encode Diagram 3](#_Toc497414626)

[4.2D Overall Use Case – Decode Diagram 3](#_Toc497414627)

[4.1 UC-01 - Encode with or without Encryption 4](#_Toc497414628)

[4.2 UC-02 - Decode with or without Encryption 5](#_Toc497414629)

[4.3 UC-03 – Advanced Options 6](#_Toc497414630)

[4.4 UC-04 - Open, View and Close Help Page 7](#_Toc497414631)

[4.5 UC-05 – Generate Fractal 7](#_Toc497414632)

[4.6 UC-05 – Open, View and Close About Page 8](#_Toc497414633)

[4.7 UC-07 – Navigation Bar/Menu System 8](#_Toc497414634)

[5. Other Nonfunctional Requirements 10](#_Toc497414635)

[5.1 Performance Requirements 10](#_Toc497414636)

[5.2 Safety Requirements 11](#_Toc497414637)

[5.3 Security Requirements 11](#_Toc497414638)

[5.4 Software Quality Attributes 11](#_Toc497414639)

[6. Other Requirements 11](#_Toc497414640)

[7. System Requirements Chart 11](#_Toc497414641)

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
| Chris Hoegger  Leilani Ray  Nathan Beyer  Chris Menning | September 22, 2017 | Create initial document | 1.0.0 |
| Nathan Beyer | October 17, 2017 | Updated project descriptions in Cover Page and Headers to “AES Encryption”, instead of “3 Way Encryption”  Update the Performance Requirements section to reflect that the software only supports Latin ASCII characters, and that our software can encode JPEG images, but cannot decode them, and the encoded image must be saved in another format | 2.0.0 |
| Nathan Beyer | October 22, 2017 | Expanded Purpose Section  Updated User Interfaces Section  Updated Software Interfaces Section | 2.0.1 |
| Nathan Beyer | October 23, 2017 | Added 3 Use Cases (U1 – U3) to Section 4 | 2.0.2 |
| Nathan Beyer | October 25, 2017 | Added 7 Use Cases (U4 – U10) to Section 4 | 2.0.3 |
| Chris Hoegger | October 30, 2017 | Created Pull Request for Version | 3.0.0 |
| Chris Hoegger | November 2, 2017 | Completed and provided updates to the following sections   1. Introduction 2. Overall Description 3. External Interface Requirements 4. Use Cases 5. Other Non-Functional Requirements 6. System Requirements – prepared example/template for use in completing section   Highlighted sections are those that need to be completed.  Highlighted verbiage indicates questions. | 4.0.0 |
| Leilani Ray | November 6, 2017 | Updated some of ChrisM comments, added some comments from ChrisH updates.  Added the letter “D” to the two diagrams to identify them uniquely from the use cases.  Worked on continuity of use cases. | 4.0.2 |
| Leilani Ray | November 8, 2017 | Incorporated all comments from this week’s review, from all team members. | 5.0 |
| Chris Menning | November 13, 2017 | Updated phrasing in Purpose section to remove erroneous sequence of events.  Added to References section.  Updated Data Flow Diagrams.  Updated UC2, UC-05. | 6.0 |

# Introduction

## Purpose

This developed software package enables users to hide messages written in plain text within bitmap images, and retrieve hidden messages from already-encoded images. The user’s message may be encoded with or without AES encryption.

## References

* [Nevon Projects: Image Steganography With 3 Way Encryption - Project idea and synopsis](http://nevonprojects.com/3-way-encryption-image-staganography/)
* [Code Project: Steganography: Simple Implementation in C#](https://www.codeproject.com/Tips/635715/Steganography-Simple-Implementation-in-Csharp)
* [Quick Crypto: Encryption information for emails and files; password handling and file shredding](http://quickcrypto.com/)
* [MSDN: Documentation of the built-in AES class for Visual Studio](https://msdn.microsoft.com/en-us/library/system.security.cryptography.aes(v=vs.110).aspx)
* [Stack Overflow: Generating AES 256 Bit Key Value](https://stackoverflow.com/questions/17195969/generating-aes-256-bit-key-value)
* [C# Helper: Draw a Mandelbrot set fractal in C#](http://csharphelper.com/blog/2014/07/draw-a-mandelbrot-set-fractal-in-c/)

# Overall Description

## User Classes and Characteristics

User(s) – Individual(s) utilizing the software to safeguard textual data through one or more available processes including encoding with optional encryption and decryption/decoding through authentication opportunities.

## Operating Environment

This software application will operate under the following systems and components:

* Windows Environment: Versions 7.0 - 10
* Thumb/USB, CD, DVD and SD drive device components

## Design and Implementation Constraints

The project team has identified the following constrains for design and implementation of this software:

* GitHub (chosen project repository and file/version management tool) learning curve
* Encryption/Decryption process understanding
* Image property and manipulation technique knowledge base

## Assumptions and Dependencies

In order to achieve the defined, team-designed, final product, it is the assumption that all team members either have a working Visual Studio application or have access as through a workstation at the College or via other means.

# External Interface Requirements

## User Interfaces

The main form includes a navigation bar at the top to assist the user in choosing functions. The availability of “About” and “Help pages along with Advanced Options (for power users) and an alternative Fractal image selection will round out the application features, creating a robust, user friendly application.

Through the design of command buttons is where the majority of user interface resides. Depending upon the situation, auto-generated prompts and/or error messages informing the user of needed next steps will appear.

Users cannot open and encode an image when its size prevents it from storing a single character or when the image size itself exceeds the size allowed by the application. This halted process would prevent the user’s ability to encode any message within the image and they should choose an alternate course.

The form’s message box dynamically limits the size of the message the user can create based on the size of the user’s chosen image. This feature, accomplished by setting the maximum character limit of the textbox to the maximum number of characters that are available to be encoded, returns a count of remaining characters. This information will be available on the screen for user reference.

Once an image is selected and message submitted, a save file dialog box is presented and the user can save their encoded image to any directory, device or location of their choosing.

The user also has available to them an optional feature. In order to secure the document image further, they have the ability to use the AES Encryption process and file’s password protection element.

Once the image is secure, the user has the option to either decode the file. The file is now ready for decryption, if necessary, and decoded so that the hidden message is displayed.

## Hardware Interfaces

There are no hardware interfaces at this time.

## Software Interfaces

The software utilizes the user’s operational system “open file” dialog box for both locating original images to be encoded, as well as the final decoded image. The window’s “save file” dialog box is triggered upon command once the image has been encoded so that the user can choose a location to place the file. Generation of both dialog windows are designed such that users can only open and save images of supported file types.

## Communications Interfaces

This application is a stand-alone product and therefore, no identified communication interface components.

# System Use Cases

## 4.1D UC-02 Data Flow Diagram

## 4.2D UC-02 Data Flow Diagram

## UC-01 - Encode with or without Encryption

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Use Case ID: | UC-01 | | | Priority | H |  | |
| Use Case Name: | **Encode with or without Encryption** | | | | | | |
| Type: | External | | X | Temporal |  |  | |
| Created By: | Chris Hoegger | | | | Last Updated By: | | Leilani |
| Date Created: | November 2, 2017 | | | | Date Last Updated: | | November 7, 2017 |
| Actors: | | User | | | | | |
| Description: | | The user encodes and saves a message into an image (with optional encryption and security features). | | | | | |
| Trigger: | | User clicks Encode button from main form | | | | | |
| Preconditions: | | 1. User has need for application | | | | | |
| Post Conditions: | | 1. User has option to decrypt and decode | | | | | |
| Normal Flow: | | 1. User chooses an image file they wish to encode 2. System opens user’s selected image file 3. User enters message into the appropriate textbox 4. System encodes message in selected image file 5. User selects where and how to save encoded image file 6. System saves encoded image file | | | | | |
| Alternative Flows: | | 1. With Optional Fractal (instead of Step 1 in Normal Flow)    1. User chooses to create fractal by clicking command    2. System generates fractal image through system sub-process    3. User continues at Step 3 in Normal Flow 2. With Optional Encryption and Password Protection (before Step 3 in Normal Flow)    1. User chooses to encrypt file checking the “Use Encryption” checkbox at any point before clicking the “Hide Text in Image” button.    2. User chooses to password protect image file    3. User continues at Step 3 in Normal Flow | | | | | |
| Exceptions: | | 1. Image is corrupt/format error    1. Exception handling generates message to user    2. User chooses new image or quits application 2. Image size is too small to contain message    1. Exception handling generates message to user    2. User chooses new image or quits application 3. Message is too large for application    1. Exception handling generates message to user    2. User revises message or quits application 4. Encryption fails    1. Exception handling generates message to user    2. User quits application 5. Password entered is not compliant with established rules or missing    1. Exception handling generates message to user    2. User enters new password 6. Selected directory or location for encoded image is not available per security measures    1. User’s system generates exception handling message    2. User options       1. changes location       2. permission levels are updated       3. File is saved | | | | | |
| Includes: | | Encoding, Encryption, Password Protection | | | | | |
| Frequency of Use: | | As often as user wishes, but process is activated once per image when encode button is clicked | | | | | |
| Business Rules: | |  | | | | | |
| Special Requirements: | |  | | | | | |
| Assumptions: | |  | | | | | |
| Notes and Issues: | |  | | | | | |

## UC-02 - Decode with or without Encryption

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Use Case ID: | UC-02 | | | Priority | H |  | |
| Use Case Name: | **Decode with or without Decryption** | | | | | | |
| Type: | External | | X | Temporal |  |  | |
| Created By: | Leilani Ray | | | | Last Updated By: | | Chris Menning |
| Date Created: | November 1, 2017 | | | | Date Last Updated: | | November 13, 2017 |
| Actors: | | User | | | | | |
| Description: | | A user wants to decode a file. | | | | | |
| Trigger: | | User clicks a button to decode the file. | | | | | |
| Preconditions: | | 1. Application is running. 2. User has access to an encoded file. | | | | | |
| Post Conditions: | | 1. The file is now decoded and shows text. 2. At this point we might want to decide what the user does with the file, encoded, decoded | | | | | |
| Normal Flow: | | 1. The user opens an already-encoded image file. 2. The user selects the “Retrieve Message from Image” button. 3. The program retrieves a hidden message from the image. 4. The retrieved text is displayed in a textbox. | | | | | |
| Alternative Flows: | | * 1. The “Use Encryption” checkbox is checked.   2. The user is prompted to enter a password.   3. Continue on from step 3. | | | | | |
| Exceptions: | | If the password is incorrect or the wrong decryption algorithm is used, the message is simply not displayed correctly. | | | | | |
| Includes: | |  | | | | | |
| Frequency of Use: | | Frequent | | | | | |
| Business Rules: | |  | | | | | |
| Special Requirements: | |  | | | | | |
| Assumptions: | | The user possesses an image bearing a hidden message. | | | | | |
| Notes and Issues: | |  | | | | | |

## UC-03 – Advanced Options

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Use Case ID: | UC-03 | | | Priority | M |  | |
| Use Case Name: | **Advanced Options** | | | | | | |
| Type: | External | | X | Temporal |  |  | |
| Created By: | Chris Menning | | | | Last Updated By: | | Chris Menning |
| Date Created: | October 29, 2017 | | | | Date Last Updated: | | October 29, 2017 |
| Actors: | | User | | | | | |
| Description: | | A power user wants to use Advanced Options for more control of the encryption process. | | | | | |
| Trigger: | | The user selects “Advanced Options” from the menu system. | | | | | |
| Preconditions: | | 1. The program is running. 2. “Use Encryption” is selected. | | | | | |
| Post Conditions: | | 1. The user supplies a password or passphrase of their choosing. The system then derives an AES key from the password/passphrase.. 2. The system remembers the user’s selected preference the next time the program is run. | | | | | |
| Normal Flow: | | 1. User selects “Advanced Options” from the menu. 2. Form presents the user with an option to select one of the following, for generating AES keys from a password:    1. Mode 1: 128 bits    2. Mode 2: 192 bits    3. Mode 3: 256 bits 3. If the user changes modes, display the following message: 4. *“WARNING: Using a custom AES block size requires that the person decrypting also use the same AES block size.  Are you sure you want to use [size]?”* 5. Encryption and decryption will now utilize the selected key size. 6. By default “Remember my preference” is checked, and the user’s preference is saved to a file and automatically reloaded next time. | | | | | |
| Alternative Flows: | | 1. User chooses not to utilize Advanced Options menu.    1. AES key set to 256 bits by default. 2. User changes AES block size, then tries to decrypt something encrypted with a different AES block Size    1. Detect scrambled output.    2. Display message: “*Something not right? Try changing AES Block Size in Advanced Options.”* | | | | | |
| Exceptions: | | None. | | | | | |
| Includes: | | UC-01, UC-02, UC-07 | | | | | |
| Frequency of Use: | | Variable | | | | | |
| Business Rules: | | AES Keys themselves must not be saved in any files. Only AES Key size preferences can be saved. | | | | | |
| Special Requirements: | | The Mode set in the Advanced Options form will be dependent upon by the Password Handler, AESEncrypt, and AESDecrypt classes. It is possible that Encoder and Decoder classes may also be affected. | | | | | |
| Assumptions: | | The term “Advanced Options” conveys that these options are not something the average user needs to mess with, but they’re present for power users that would like more control. | | | | | |
| Notes and Issues: | |  | | | | | |

## UC-04 - Help Page

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Use Case ID: | UC-04 | | | Priority | High | |  | |
| Use Case Name: | **Help Page** | | | | | | | |
| Type: | External | | X | Temporal |  |  | | |
| Created By: | Nathan Beyer | | | | Last Updated By: | | | Leilani |
| Date Created: | October 31, 2017 | | | | Date Last Updated: | | | November 8, 2017 |
| Actors: | | User | | | | | | |
| Description: | | Details the user opening the Help page from the main form | | | | | | |
| Trigger: | | User clicks a button on the main form to open the Help page | | | | | | |
| Preconditions: | | 1. Application is open | | | | | | |
| Post Conditions: | | 1. User has opened and viewed the Help page | | | | | | |
| Normal Flow: | | 1. User clicks a button on the main form to open the Help page 2. System displays the Help page 3. User closes Help page when finished viewing it by clicking the Close button | | | | | | |
| Alternative Flows: | | none | | | | | | |
| Exceptions: | | an exception is thrown when the Main Form tries to open the Help Page”, as that is a possible (though incredibly unlikely) exception | | | | | | |
| Includes: | | None | | | | | | |
| Frequency of Use: | | High | | | | | | |
| Business Rules: | |  | | | | | | |
| Special Requirements: | |  | | | | | | |
| Assumptions: | |  | | | | | | |
| Notes and Issues: | |  | | | | | | |

## UC-05 – Generate Fractal

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Use Case ID: | UC-05 | | | Priority | High |  | |
| Use Case Name: | **Generate Fractal** | | | | | | |
| Type | External | | X | Temporal |  |  | |
| Created By: | Nathan Beyer | | | | Last Updated By: | | Chris Menning |
| Date Created: | November 2, 2017 | | | | Date Last Updated: | | November 13, 2017 |
| Actors: | | User | | | | | |
| Description: | | A user has a message that they want hidden in an image, but they don’t have a particular image in mind. They should be given the option to generate a fractal. | | | | | |
| Trigger: | | User clicks “Generate Fractal” menu item or button. | | | | | |
| Preconditions: | | 1. Application is open | | | | | |
| Post Conditions: | | 1. A fractal image is saved with the image hidden inside. | | | | | |
| Normal Flow: | | 1. The user selects “Generate Fractal” from the File menu, or clicks a “Generate Fractal” Button. 2. The picture box displays a fractal. 3. The user may continue with UC-01. | | | | | |
| Alternative Flows: | | 1. The “Use Encryption” option is checked. 2. See Alternative Flow for UC-01. | | | | | |
| Exceptions: | |  | | | | | |
| Includes: | | UC-01 | | | | | |
| Frequency of Use: | | Variable | | | | | |
| Business Rules: | |  | | | | | |
| Special Requirements: | |  | | | | | |
| Assumptions: | |  | | | | | |
| Notes and Issues: | |  | | | | | |

## UC-06 – About Page

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Use Case ID: | UC-06 | | | Priority | High |  | |
| Use Case Name: | **About Page** | | | | | | |
| Type: | External | | X | Temporal |  |  | |
| Created By: | Nathan Beyer | | | | Last Updated By: | | Leilani |
| Date Created: | October 29, 2017 | | | | Date Last Updated: | | November 8, 2017 |
| Actors: | | User | | | | | |
| Description: | | Details the user opening the About page from the main form | | | | | |
| Trigger: | | User clicks a button on the main form to open the About page | | | | | |
| Preconditions: | | 1. Application is open | | | | | |
| Post Conditions: | | 1. User has opened and viewed the About page | | | | | |
| Normal Flow: | | 1. User clicks a button on the main form to open the About page 2. System opens the About page 3. System fills the textbox on the About page with the information stored inside the project’s readme file 4. User closes About page when finished viewing it | | | | | |
| Alternative Flows: | | 1. System cannot find the readme file or read text from it    1. System displays an appropriate error message in a message box    2. System places the error message inside the textbox on the About page. | | | | | |
| Exceptions: | | System encounters error locating or reading text from the project’s readme file | | | | | |
| Includes: | | None | | | | | |
| Frequency of Use: | | High | | | | | |
| Business Rules: | |  | | | | | |
| Special Requirements: | |  | | | | | |
| Assumptions: | |  | | | | | |
| Notes and Issues: | |  | | | | | |

## UC-07 – Navigation Bar/Menu System

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Use Case ID: | UC-07 | | | Priority | Medium |  | |
| Use Case Name: | **Menu System** | | | | | | |
| Type: | External | | X | Temporal |  |  | |
| Created By: | Chris Menning + ChrisH | | | | Last Updated By: | | Leilani |
| Date Created: | October 31, 2017 | | | | Date Last Updated: | | November 6, 2017 |
| Actors: | | User | | | | | |
| Description: | | Menu options for user choice. Many users find the familiarity of a top-bar menu with File, Edit, Help, etc to be an intuitive means of using an application. | | | | | |
| Trigger: | | The user clicks one of the following:   * File * Edit * Preferences * Help | | | | | |
| Preconditions: | | 1. The program is running | | | | | |
| Post Conditions: | | 1. The user can locate and execute their desired command. | | | | | |
| Normal Flow: | | 1. The user selects the File menu.    1. The following items appear in a dropdown menu.       1. Open Un-encoded Image          1. (see UC-01)       2. Save Encoded Image (initially disabled)          1. (see UC-01)       3. Generate New Un-encoded Image          1. See (UC-05)       4. Open Encoded Image          1. (see UC-02)       5. Save Decoded Message (initially disabled)          1. Open a Save dialog and output a text file.    2. The user clicks one of the listed items and the command executes. 2. The user selects the Edit menu.    1. The following items appear in a dropdown menu.       1. Cut       2. Copy       3. Paste       4. Delete       5. Select-All    2. The user clicks on of the listed items, and the edit is executed on the text in either the input or output textboxes, whichever had focus last. 3. The user selects the Preferences menu.    1. The following items appear in a dropdown menu.       1. Advanced Options          1. (see UC-07) 4. The user selects the Help menu.    1. The following items appear in a dropdown menu.       1. View Help          1. (See UC-04)       2. About          1. (See UC-06) | | | | | |
| Alternative Flows: | | 1. The user selects the Edit menu.    1. The cursor was not on a textbox, so all items in Edit menu are disabled. | | | | | |
| Exceptions: | | None. | | | | | |
| Includes: | | UC-01, UC-02., | | | | | |
| Frequency of Use: | | Frequent | | | | | |
| Business Rules: | |  | | | | | |
| Special Requirements: | |  | | | | | |
| Assumptions: | | The most general use cases from the average user can be accomplished without the menu system. But items like user preferences, help, and the about screen may be better organized using the familiar menu system interface. | | | | | |
| Notes and Issues: | |  | | | | | |

# Other Nonfunctional Requirements

## Performance Requirements

This application only supports American Standard Code for Information Interchange (ASCII) characters for message encoding. As a result, English is the only supported language. Unicode is not supported.

Virtually any popular rasterized image format may be used for input, but the resulting image will always save as a PNG.

System performance is noticeably slower when choosing to use encryption while hiding messages as compared to hiding messages without encryption. This is normal.

**In this section, just say “See section 7 requirements 23-27”. And I’ll assume those requirements are Performance related.**

## Safety Requirements

This software is available in an “as is” condition, meaning that utilization should be at operator’s own risk and caution exercised regarding safeguarding of confidential information as well as choice of image files used.

## Security Requirements

There are no guarantees that information entered into this software will be 100% safeguarded against possible decryption from third party use.

## Software Quality Attributes

<Specify any additional quality characteristics for the product that will be important to either the customers or the developers. Some to consider are: adaptability, availability, correctness, flexibility, interoperability, maintainability, portability, reliability, reusability, robustness, testability, and usability. Write these to be specific, quantitative, and verifiable when possible. At the least, clarify the relative preferences for various attributes, such as ease of use over ease of learning.>

**In this section, just say “See section 7 requirements 55-62”. And I’ll assume those requirements are Software Quality related.**

# Other Requirements

<Define any other requirements not covered elsewhere in the SRS. This might include database requirements, internationalization requirements, legal requirements, reuse objectives for the project, and so on. Add any new sections that are pertinent to the project.>

You may not have any.

# System Requirements Chart

*< Include a* ***table*** *in this section with the following columns:*

***ID*** *– Unique requirement ID*

***Priority*** *– Priority of this requirement*

***Type*** *– Functional(F) or Non-functional(NF)*

***Source*** *– Who is most interested in this requirement (John Smith – Customer). For this project you can make it up, in reality you’ll want to capture this as you capture the requirements.*

***Contained in Use Case(s****) – Which use cases reference this requirement or which use cases when executed will perform this requirement. There may be a few functional requirements without a use-case and the non-functional requirements generally will NOT be part of a use-case (so put N/A).*

***Description*** *– The description of the requirement. “The system shall …. “*

*>*

These requirements should match up with your use case diagrams.

Example chart for purpose of template. SRS example (with minor revisions) can be found at: <https://cs.gmu.edu/~dfleck/classes/cs421/spring08/SampleProject/FINAL%20SRS.pdf>

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Priority** | **Type \*** | **Source** | **Use Case(s)** | **Description** |
| 01.0.0 | High | F | Customer- John Doe | UC-01 | The user encodes and saves a message into an image |
| 01.0.1 | High | F | Customer- John Doe | UC-01 | User chooses an image file they wish to encode |
| 01.0.2 | High | F | Customer-John Doe | UC-01 | System opens user’s selected image file |
| 01.0.3 | High | F | Customer- John Doe | UC-01 | User enters message into the appropriate textbox |
| 01.0.4 | High | F | Customer-John Doe | UC-01 | System encodes message in selected image file |
| 01.0.5 | High | F | Customer-John Doe | UC-01 | User selects where and how to save encoded image file |
| 01.0.6 | High | F | Customer-John Doe | UC-01 | System saves encoded image file |

\*Type

NF=Nonfunctional

F=Functional