|  |  |
| --- | --- |
| http://paperpkblog.com/admissions/wp-content/uploads/2015/05/Jobs-in-Foundation-University-Islamabad-FUI-20141.jpg | **FOUNDATION UNIVERSITY ISLAMABAD** |

**ALHAZEN IMAGE FORENSICS SYSTEM**

***Submitted by***

F-18-BCSE-007 – Bilal Fawad Khan

F-18-BCSE-014 – Humna Hayee

F-18-BCSE-161 – Raheel Kiany

F-18-BCSE-169 – Syed Abis Abbas Zaidi

**BACHELORS**

**IN**

**COMPUTER SOFTWARE ENGINEERING**

**YEAR**

**2018-2022**

***Project Advisor and Coordinator***

Dr. M. Ishtiaq

Submitted to the Department of Software Engineering of Foundation University Islamabad, in partial fulfilment for the requirements of a Bachelor of Computer Degree in Software Engineering.

**DEDICATION**

We dedicate our work to our beloved parents who have been great support to us through our educational journey, our teachers who guided us through this transitional time. Without their support and help, we might never have achieved our goals. We also dedicate this project to our project advisor and coordinator Dr. M. Ishtiaq who provided us with the best kind of knowledge and guidance. It would be impossible to achieve our goals without all of you.

**ACKNOWLEDGEMENTS**

Everyone has an important role in your success, final year project is an important milestone in bachelor’s degree of any student broadening his/her vision and depicting the upcoming challenge in real world. This is a man’s world where we help, support guide, motivate to achieve each other’s goals. Final year project was one of the challenging projects in BCSE degree, and we cannot forget mentioning Dr. M. Ishtiaq who has supported and helped us to complete this project.

**STATEMENT OF ORIGINALITY**

We namely **Bilal Fawad Khan, Humna Hayee,** **Raheel Kiany,** and **Syed Abis Abbas Zaidi, s**ubmitting thesis titled “**Alhazen Image Forensics System**”. State that we clearly understand what plagiarism is and we have also read about it using online sources. As such, we claim that this entire work (code, reports etc.) is our own effort. We have not used/copy or pasted or paraphrased even a single line of code or sentence from any other source without giving a proper reference to it. Based on our confidence in our originality, we allow the university to run anti-plagiarism software on the same.

We understand that if now, or in the future, ever it is found out that we have not been honest and have plagiarized, the university will take action and will result in disciplinary action, such as the cancellation of our degree and we can be liable for any subsequent punishments as deemed appropriate by the University as well as the Higher Education Commission Pakistan.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Bilal Fawad Khan

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Humna Hayee

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
Raheel Kiany

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Syed Abis Abbas Zaidi

**COPYRIGHT STATEMENT**

Copyright in text of this report rests with the student authors. Copies (by any process) either in full, or of extracts, may be made only in accordance with instructions given by the author. Details may be obtained by the Librarian. This page must form part of any such copies made. Further copies (by any process) of copies made in accordance with such instructions may not be made without the permission (in writing) of the author.

The ownership of any intellectual property rights which may be described in this project is vested in FUI, subject to any prior agreement to the contrary, and may not be made available for use by third parties without the written permission of the FUI, which will prescribe the terms and conditions of any such agreement.

**CERTIFICATE OF APPROVAL**

It is certified that Project titled **“Alhazen Image Forensics System”,** presented on

4-July-2022, has been duly approved by the evaluation committee.

|  |  |
| --- | --- |
| **Project Advisor:** | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Dr. M. Ishtiaq |
| **Project Coordinator:** | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Dr. M. Ishtiaq |
| **Manager Graduate Off:** | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Engr. Sherjeel Farooqui |

**ABSTRACT**

In today's world of internet, information spreads in a matter of seconds. The connectivity offered by social media has revolutionized our personal, economic, and social life. People have access to information with a single click, which also has its pros and cons. Many people/organizations use this quick access to the internet for their propaganda purpose, they spread their version of events and/or views about an event, idea or entity and using digital media is a major way of doing that as people tend to believe more on visual proofs.

Our project focuses on providing its users to check for traces of forgery in the subject images. This ensures that users such as academic professionals, researchers of the digital media forensics domain, law enforcement agency professionals, journalists and other media professionals, and the public should have a tool available to filter out forged images to increase their work’s efficiency and credibility.

**TABLE OF CONTENTS**

**Chapter 1: Introduction ….……………………………………………………… 1**

* 1. Introduction ………………………………………………………………... 1
  2. Existing System …………………………………………………………… 1
  3. Literature Review …………………………………………………………. 2
  4. Problem Definition ………………………………………………………... 3
  5. Context Diagram …………………………………………………………... 3

**Chapter 2: Introduction to Proposed System ………………………………….. 4**

* 1. Introduction ……………………………………………………………….. 4
  2. Project Background ……………………………………………………….. 4
  3. Problem Description ………..……………………………………………... 4
  4. Project Objectives …………..……………………………………………... 5
  5. Project Scope ……………….……………………………………………... 6
     1. Constraints …………………………………………………………..…….. 6
     2. Deliverables ………………………………………………………..……… 6
  6. Project Features ……………..…………………………………………….. 6
     1. Image Analysis ……………………………………………………………. 6
     2. Tamper Detection …………………………………………………………. 6
     3. Batch Processing ………………………………………………………….. 6
     4. Export Result ……………………………………………………………… 6

**Chapter 3: Requirements Specifications ...……………………………………... 7**

* 1. Introduction ………………...……………………………………………... 7
  2. Functional Requirements …….….…...……………..................................... 7
  3. DFD Level 1 ………………..……………………………………………... 8
  4. DFD Level 2 ………………..……………………………………………... 9
  5. Graphical User Interface …………………………………………………... 10
  6. Non-functional Requirements ……………………………………………... 11
     1. Safety Requirements ………………………………………………………. 11
     2. Performance Requirements ………………………………………….…….. 11
     3. Security Requirements ……………………………………………..……… 12
     4. Useability …………………………………………………………..……… 12
     5. Availability ………………………………………………………….…….. 12
     6. Maintainability ……………………………………………………………. 12
     7. Reliability …………………………………………………………………. 12
     8. Accuracy ………………………………………………………………...… 12

**Chapter 4: Design Specifications ………………………………………………... 13**

1. Introduction …………..…….……………………………………………... 13
2. System Architect ..………………….……………....................................... 13
3. Design Methodology ……………………………………………………… 14
4. High Level Design ……………...……………………………………….... 14
5. Data Design ……………………..………………………………………… 15

**Chapter 5: Test Specifications …………………………………………………... 16**

1. Introduction ……………………………………………………………...... 16
2. Test Design …………………….…………………...................................... 16
   * 1. White Box Testing ……………………………………………………….... 16
     2. Black Box Testing ………...……………………………………………..... 16
     3. GUI Test Cases ……………………………………………………………. 23
     4. Useability Testing …………..……………………………………………... 25
     5. Software Performance Testing .…………….……………………………... 25
     6. Compatibility Testing ………..……………………...…………………….. 25
     7. Installation Testing ………..………………………...…………………….. 26
     8. Acceptance Testing ……………………………………………………….. 26
3. Defect Report ………….……...…………..………………………………. 26
4. Test Report ………..………...…………………………………………….. 28

**Chapter 6: Conclusion …….……………………………………………………... 29**

1. Introduction ……………………………………………………………….. 29
2. Conclusion …………………….…………………...................................... 29
3. Benefits ………………………………………………………………...…. 29
4. Future Work ……………….……………………………………………... 29

**Bibliography …………………..…………………………………………….......... 30**

**LIST OF TABLES**

**[Table 1.1:](#_Toc449345721)** [Benchmarking 1](#_Toc449345721)

[**Table 3.1:** Functional Requirements 7](#_Toc449345721)

**[Table 5.1:](#_Toc449345721)** [Iniating AIFS 17](#_Toc449345721)

[**Table 5.2:** Image Upload 1](#_Toc449345721)8

[**Table 5.3:** Metadata Extraction 19](#_Toc449345721)

[**Table 5.4:** Histogram Display 20](#_Toc449345721)

[**Table 5.5:** Forgery Detection 21](#_Toc449345721)

[**Table 5.6:** Export Results 22](#_Toc449345721)

[**Table 5.7:** Initiating AIFS GUI 23](#_Toc449345721)

[**Table 5.8:** Image Upload GUI 23](#_Toc449345721)

[**Table 5.9:** Metadata Extraction GUI 24](#_Toc449345721)

[**Table 5.10:** Histogram Display GUI 24](#_Toc449345721)

[**Table 5.11:** Software Performance Testing 25](#_Toc449345721)

[**Table 5.12:** Compatibility Testing 25](#_Toc449345721)

[**Table 5.13:** Installation Testing 26](#_Toc449345721)

[**Table 5.14:** Acceptance Testing – Alpha 26](#_Toc449345721)

[**Table 5.15:** Acceptance Testing – Beta 26](#_Toc449345721)

[**Table 5.16:** Defect Report – Image Upload 27](#_Toc449345721)

[**Table 5.17:** Defect Report – Metadata Extraction 27](#_Toc449345721)

[**Table 5.18:** Test Report 28](#_Toc449345721)

**LIST OF ILLUSTRATIONS**

[**Figure 1.1:** Context Diagram 3](#_Toc449345672)

[**Figure 3.1:** DFD Level 1 8](#_Toc449345672)

[**Figure 3.2:** DFD Level 2 9](#_Toc449345672)

[**Figure 3.3:** Image Uploading Interface 10](#_Toc449345672)

[**Figure 3.4:** Metadata Extraction Interface 11](#_Toc449345672)

[**Figure 4.1:** System Architecture Representation](#_Toc449345672) 13

[**Figure 4.2:** High Level Design 14](#_Toc449345672)

**LIST OF ABBREVIATIONS**

|  |  |
| --- | --- |
| **Abbreviations** | **Description** |
| LEA | Law Enforcement Agencies |
| AIFS | Alhazen Image Forensics System |
| DFD | Data Flow Diagram |
| SEM | Scanning Electron Microscope |
| GUI | Graphical User Interface |
| GPU | Graphics Processing Unit |
| API | Application Programming Interface |
| NFR | Non-Functional Requirement |
| TBD | To be determined |
| N/A | Not applicable |
| OS | Operating System |

Chapter 1

# INTRODUCTION

* 1. **Introduction**

This report contains a brief description of our project AIFS, it covers the complete details about how the project will be completed with all the necessary diagrams. The first chapter discusses the existing system related to our project, problem definition and literature review. The second chapter gives the background and overview, project scope, objectives, features, and use case diagram. The third chapter describes the requirement specifications which will consist of the functional and non-functional requirements, and other diagrams related to the understanding of the project.

* 1. **Existing System**

A proper education of the problem is very important for all of us in-order to solve the problem. In the current situation, people of our society are facing a lot of difficulties especially when it comes to justice. Most of the underprivileged people are trapped into fake image scam by upper class for crimes in which they are not involved and court sentence them punishment for what they are not guilty. By creating this software, the purpose is to facilitate the poor people of our society with a system to prove them innocent.

The Table 1.1 provides a comparison of our product with a few existing softwares that have been launched previously in order to detect forgery in image.

**Table 1.1:** Benchmarking

|  |  |  |  |
| --- | --- | --- | --- |
| **Features** | **Amped Authenticate** | **SEM** | **AIFS** |
| **Upload image** | ✓ | ✓ | ✓ |
| **Display Metadata** | ✓ | × | ✓ |
| **Display Histogram** | × | × | ✓ |
| **System Offline** | × | × | ✓ |
| **Unpaid software** | × | × | ✓ |
| **Photo Comparison** | ✓ | × | × |

* 1. **Literature Review**

Technology is playing key role in 21st century and is getting advance every day, so are the problems cause by it. We need to work on these problems to help people of our society. Pakistan owns the advance hardware, but they don't own quality software to solve such problems. Almost every person is holding camera and an image is just one click away. So, in front of human witness everyone gives importance to an image as a person can only describe the situation but with the help of an image, they can view the situation. To solve this issue, we have developed a software named “Alhazen Image Forensics System” which is a personal platform that helps to detect forgery in an image. AIFS is a software that will facilitate with different image forgery detection methods. It will provide image histogram and metadata result after detection of image.

The AIFS is developed based on the research carried out by Haodong Li (Student Member of IEEE), Weiqi Luo (Senior Member of IEEE) and Jiwu Huang (Fellow of IEEE) which was published in [2].

Over the years the fake images in our county are making situation worst for many people. Due to various reasons has pushed many people into a mode of life of struggle for survival especially for the people who were trapped. This application will help many to get out from such traps and live a respectful life by proving themselves not guilty.

In first step user will upload an image. In next step user will apply a method of his choice to detect a forgery. In third step system will detect forgery in the image provided by the user. In next step user can view the result provided by the system, if there is a forgery user can request to view the metadata and histogram. AIFS is researched based project.

The development of AIFS begins with an introductory stage, during which it is necessary to understand the domain area, identify the target audience, determine the development tools, and calculate the project parameters (cost, capacity, quality, and duration). At the end of this stage, a project-plan should be prepared. Then the organizational stage begins – Designing the interface, panels for image detection providing basic facilities and organizing the features. The next phase is Management which implies marketing. The last stage is to launch. This software will help to identify the difference between real and fake image and decrease the number of problems causing by the fake images by providing detailed results.

* 1. **Problem Definition**

In today’s world it is difficult to identify the difference between real and a fake image. People of our society faces many issues just because of fake images. Furthermore, there is no software available where they can check either the image provided at various places as evidence is fake or real. Justice is not given to the innocent. Even large origination is manipulated with the help of image. There is no such specific software which can solve all the problems society is facing. There is a need for system that provides all the facilities in a single application, which provide them ease and facilitate them in various cases. Moreover, other software’s are not reliable and there is a chance of embezzlement to avoid this issue this project is developed.

* 1. **Context Diagram**

Diagram

Description automatically generated

**Figure 1.1:** Context Diagram

Chapter 2

# INTRODUCTION TO PROPOSED SYSTEM

* 1. **Introduction**

In this chapter the readers will be able to have an idea about the project idea and an overview of the whole project to understand what the project is about. Then the problem description and project objectives will describe what problems are there for which the project is being developed and how the project addresses those specific problems. Project objectives and scope will give a better explanation of the importance of this project and the overall scope of this project. Lastly the feature list in this chapter will highlight the features of the project and then the use case will give a better understanding of the functionalities user will be able to have from the project.

* 1. **Project Background**

The project is being developed in a time when there is misinformation, fake news, fraud, and propaganda is a common problem for everyone from a common man to even the governments of the countries. Not just the underdeveloped or developing countries for that matter but rather even all the developed countries are also suffering from the issue mentioned above.

Our project has resulted in enabling its addressed users to tackle with one of the most used misleading elements nowadays, images.

* 1. **Problem Description**

Nowadays all types of people come across a point when their professional or personal life depend on making decisions, decisions to form and opinions which ultimately lead people in taking a certain course of action.

Now the problem is that the opinion is made based on some evidence which can be an image being circulated on media, an image submitted to journalist as a proof for a news story, an image submitted as an academic work to a teacher by a student, an image being presented in the court of law. In short, that person must make a certain opinion based on that image and then must take a certain course of action, but that person does not have any means to verify the authenticity of that image.

There is no publicly available easy-to-use tool to detect the image has been tampered or not. In such circumstances any course of action taken will always have the shadow of doubt over it.

We have seen and, in some cases, heard of many celebrities defamed and mentally stressed by trolls, public office holders ridiculed and trolled, newspapers retracting their news articles after finding out that their “visual proofs” were not authentic, and students submitting ingenuine academic work and what not. All that is because of some images not being able to be checked for authenticity because there is simply no publicly available tool in Pakistan to enable users to do so.

Our AIFS will overcome the issues mentioned above by providing a simple way to users to detect traces of forgery in images.

* 1. **Project Objectives**

Following will be the intended objectives of the project:

* Enabling users to view the metadata information for their desired image such as image creation date and time, geotag location of image, device and model information of the image etc. and all that with no requirements of any technical expertise.
* Enabling user to view the visual elements data or to save that data (in case of batch image processing) such as image colour density, colour saturation, and texture etc. within few seconds.
* To give user the ability to view the histogram of the subject image with high accuracy.
* To enable user to have an easy-to-use publicly available tool for checking for traces of forgery in subject images.
* Proving users with a system that does not require any specialized training but rather should be useable for every common person aware of a computer.
  1. **Project Scope**
     1. **Constraints**
* AIFS will be developed and useable in a Windows operated machine only.
* AIFS will address still photo images only, no other types of media file formats will be covered by it.
* No corrupt image files will be processible by the AIFS.
* AIFS will need a minimum of 8GB RAM space for smooth working.
* Desktop machine should have a GPU for image processing.
  + 1. **Deliverables**
* AIFS will be the first of its kind system publicly available for users.
* It will be a desktop-based system which will provide its user and interactive and easy-to-user interface to adapt to thus the AIFS will be useable with almost every type of user who is aware of how a computer works.
* The user will not have to download any pre-requisite external component of AIFS, it will be a stand-alone system with only the conventional I/O device.
* It will process images in the least time possible but can take relatively longer duration to process large size images.
  1. **Project Features**

The features of AIFS are following:

* + 1. **Image Analysis**

When the image is uploaded, the system will extract image file data from it such as file size, creation date, orientation etc. visual elements data such as colour density and texture etc.

* + 1. **Tamper Detection**

The system will run the algorithm on the uploaded image to check whether it has been tampered or not.

* + 1. **Batch Processing**

The system will read the user’s uploaded images for image file data and image visual data and then apply the algorithm on them.

* + 1. **Export Result**

The result of the image batch image analysis and the forgery detection algorithm will be exported to a text document or a CSV file.

Chapter 3

# REQUIREMENTS SPECIFICATIONS

2. 1. **Introduction**

In this chapter the readers will be able to have an idea about the requirements specification and after reading this section of the document the readers will be able to understand the dos and don’ts of the system. This includes functional requirements which are the functions that the system must perform such as metadata extraction, forgery detection etc. which are briefly described below. Then the data flow diagrams will help the reader understand that how the data flows in the system from image uploading to forgery detection and displaying of results, for this, DFD level 1 and level 2 are given. Then the screenshots of the user interface under the GUI will help the user to get to know how the user interface will loPass like and lastly at the end the non-functional requirements are listed which will show the system’s operation capabilities and constraints on its functionality.

* 1. **Functional Requirements**

**Table 3.1:** Functional Requirements

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Features** | **REQ-1** | **REQ-2** | **REQ-3** | **REQ-4** |
| **Image Analysis** | Image should be in specific format | Image should not be corrupted. | N/A | N/A |
| **Tamper**  **Detection** | Image should be in a specific format | Image should not be corrupted. | Image must be in RGB or BG | N/A |
| **Batch**  **Processing** | Image should be in a specific format | Image should not be corrupted. | Image must be in RGB or BG | There should be more than one image in the directory |
| **Export**  **Results** | Result should be available | Image should be present in the project directory. | N/A | N/A |

* 1. **DFD Level 1**

A level 1 DFD is more detailed than a level 0. It breaks down the main processes into sub processes that can then be analysed and improved on a more intimate level. Level 1 DFD’s aim to give an overview of the full system. They will explain the system in more detail. Major processes are broken down into sub-processes. Level 1 DFD also identifies data stores that are used by the major processes. Circle represents the process while rectangles contain the externals entities.

Diagram

Description automatically generated

**Figure 3.1:** DFD Level 1

* 1. **DFD Level 2**

**Diagram

Description automatically generated**

**Figure 3.2:** DFD Level 2

* 1. **Graphical User Interface**

When the user opens AIFS this page is the first thing they will see. From here they can select the desired image by clicking on the plus (+) icon.

**A screenshot of a video game

Description automatically generated**

**Figure 3.3:** Image Uploading Interface

1. On this screen, the user will have to upload the subject image.
2. The screen will be having minimal control elements to ensure ease of understanding for the users.
3. This is a crucial interface, as the user would not be able to use the functionality of the product system without uploading the subject image file on the interface.
4. If the image file type is unsupported, the user will be prompted and asked to upload a compatible file type.

**Graphical user interface

Description automatically generated**

**Figure 3.4:** Metadata Extraction Interface

1. On this screen, the user will have to upload the subject image.
2. The screen will be having minimal control elements to ensure ease of understanding for the users.
3. This is a crucial interface, as the user would not be able to use the functionality of the product system without uploading the subject image file on the interface.
4. If the image file type is unsupported, the user will be prompted and asked to upload a compatible file type.
   1. **Non-functional Requirements**
      1. **Safety Requirements**

The system should be reliable if any error occurs then it should maintain itself within minimum time. System should save important data in case of sudden termination or problem in the system.

* + 1. **Performance Requirements**

Response time of AIFS is how much time it requires to detect the Forgery in the image/images provided. It depends on the user’s computer, as the system performs heavy computing, so it requires a dedicated graphical processing unit. By using the recommended system, the AIFS must perform its operations under 15 seconds.

* + 1. **Security Requirements**

Security of the system is not very important for users as the application is standalone and does not require any interference form a third-party app or API so there is not any chance of data leakage.

* + 1. **Useability**

AIFS has a simple graphical user interface. Any warning or error messages created by the application will be clear and brief. In case of GPU unavailability, the system will automatically shift to perform all the computations on the processor.

* + 1. **Availability**

The system should be available 24/7, it should be able to deliver the service at any time.

* + 1. **Maintainability**

The system shall follow material design guidelines and shall be designed such a way that it can be easily maintained in the future.

* + 1. **Reliability**

The system will be sufficiently powerful to have a high level of adaptation to internal failure. The system will have the capacity to recover from errors.

* + 1. **Accuracy**

The accuracy is to provide the right and correct results. The AIFS will be using most accurate and advanced algorithms which are proven and tested.

Chapter 4

# DESIGN SPECIFICATIONS

* 1. **Introduction**

In this chapter, we will emphasize the design and architecture of AIFS. The purpose of this chapter is to make developers understand the whole design of our system. Design Specifications involves several procedures during the implementation of the project. This section defines the full specifications of the project. The specifications of the project have been brPassen into smaller categories that will describe all the elements of the project design. It details the requirements that must be met for the project or process to be successful. It will also describe the project operational environment, project design limitations, design issues, modules and dependencies, language constraints, and domain constraints.

* 1. **System Architect**

The given figure explains the system architecture of our system i.e. how data is represented to the user and is processed within the system modules and how it is exported in file within the system.

Diagram

Description automatically generated

**Figure 4.1:** System Architecture Representation

* 1. **Design Methodology**

We have developed AIFS with structured design approach, this approach is the conceptualization of problem into several well organized elements of solution. As AIFS is providing its overall functionality by performing subdivided tasks through different modules i.e. module for image upload, for image metadata extraction, for histogram display, for forgery detection, for batch processing and for exporting results into file. The small pieces of problem are solved by means of solution modules. Structured design emphasis that these modules be well organized in order to achieve precise solution. All of the modules mentioned above are arranged in hierarchy and communicate with each other to provide a complete package of functionality of our system.

* 1. **High Level Design**

In the high level design of the system a more detailed system architecture illustration is provided below that give more in-depth insights into the inter-modular communications between the different modules of the system.

Diagram

Description automatically generated

**Figure 4.2:** High Level Design

* 1. **Data Design**

Since AIFS is a standalone application hence there is no requirement of any internet connectivity or any database however it does process images and export result in file storage within the system in the form of CSV files.

Chapter 5

# TEST SPECIFICATIONS

* 1. **Introduction**

This chapter is a high-level overview defining our testing strategy for AIFS. Its objective is to communicate project-wide quality standards and procedures. This chapter of the report contains the documentation of the test phase of the project. You will be seeing outcomes of different test strategies applied on the system such as unit testing, integrartion testing and system testing coverred under the black box testing strategy, then this chapter would have documentation of white box testing, GUI testing, testing of other NFRs, useability testing, and compatibility testing. Then the chapter will have the defect sheet along with the test report.

* 1. **Test Design**

At this stage of working on our project AIFS, we deduced different test cases for testing the system and outlined the structure of testing activities for the project. We developed this test design to portray the test effort, in order to give project and test personnel a framework on the boundary and scope of the test program.

We created the test design by following the following steps:

* Read and analyzed the requirements
* Noted down the prerequisite and required informations steps
* Wrote the procedure such as steps to perform configurations etc.
* Wrote the steps to capture the tester input for objective evidence.
  + 1. **White Box Testing**

The internal functionality of the product is tested with the help of white box testing. In this process, all the running programs are tested in detail to identify any logical errors, bugs, or system loopholes that are hard to detect if testing is not due and can causes damages. This testing assisted us in improving designs and enhancing protection.

* + 1. **Black Box Testing**

The external functionality of the product is tested with the help of black box testing without any knowledge of its internal functionality, code structure, or implementation details. The black box testing tests how the system functions for an end-user.

|  |  |  |
| --- | --- | --- |
| **TEST CASE ID** | **TC:1.0** | |
| **Test Case Name** | Initiating AIFS | |
| **Purpose** | To check the functionality of application when clicking on app icon front window opens. | |
| **Pre-Condition** | The user must have AIFS installed. | |
| **Post-Condition** | The user has already installed AIFS. | |
| **Scenario** | **User action** | **System response** |
|  | Users install the application provided by the developer in the device and click the icon to open the application. | If user installs the application in the system. System check it is installed successfully. System should open the installed app windows. |
|  | The user didn’t install the application properly, user will reinstall the application. | The system check app is installed properly. If app is not installed properly it doesn’t open it. |
| **Actor** | User | |
| **User Input** | User clicks the icon of AIFS. | |
| **Expected Result** | The system opens the front window of AIFS . | |
| **System Output** | The system opens the front window of AIFS. | |
| **Status** | Pass | |
| **Created By** | Humna Hayee & Raheel Kiany | |
| **Executed By** | Humna Hayee & Raheel Kiany | |
| **Date** | 1st May, 2022 | |

**Table 5.**1**:** Initiating AIFS

|  |  |  |
| --- | --- | --- |
| **TEST CASE ID** | **TC:1.1** | |
| **Test Case Name** | Image Upload | |
| **Purpose** | To check the functionality of image uploading. | |
| **Pre-Condition** | Image exist in the system. | |
| **Post-Condition** | Image is uploaded by the user for forgery detection. | |
| **Scenario** | **User Action** | **System Response** |
|  | After front screen appears on the windows of AIFS. User selects the option of please upload the image first. | The button will allows user to enter in the system and select the following image. |
|  | User will select the image and upload it for the forgery detection in AIFS. | If the image file type is unsupported, the user will be prompted and asked to upload a compatible file type |
| **Actor** | User | |
| **User Input** | Uploads image from the system. | |
| **Expected Result** | System will show an image on AIFS windows after users uploaded the image. | |
| **System Output** | System displayed an image on AIFS windows after users uploaded the image. | |
| **Status** | Pass | |
| **Created By** | Humna Hayee & Raheel Kiany | |
| **Executed By** | Humna Hayee & Raheel Kiany | |
| **Date** | 1st May, 2022 | |

**Table 5.2:** Image Upload

|  |  |  |
| --- | --- | --- |
| **TEST CASE ID** | **TC:1.2** | |
| **Test Case Name** | Metadata Extraction | |
| **Purpose** | To extract the meta data of the image user uploaded for detailed information. | |
| **Pre-Condition** | Image should be present in the AIFS. | |
| **Post-Condition** | Metadata tags will be visible in dedicated GUI elements. | |
| **Scenario** | **User Action** | **System Response** |
|  | User clicks the MT option on the screen, to view image meta-data. | The system will read the image and extract the meta-data from the image to display it on the screen. |
| **Actor** | User | |
| **User Input** | User will click on option of MT. | |
| **Expected Result** | The user can see the result of metadata extracted by the system on screen. | |
| **System Output** | The meta-data will be displayed. | |
| **Status** | Pass | |
| **Created By** | Humna Hayee & Raheel Kiany | |
| **Executed By** | Humna Hayee & Raheel Kiany | |
| **Date** | 1st May, 2022 | |

**Table 5.3:** Metadata Extraction

**Table 5.4:** Histogram Display

|  |  |  |
| --- | --- | --- |
| **TEST CASE ID** | **TC:1.3** | |
| **Test Case Name** | Histogram Display | |
| **Purpose** | To check the image grey level detail in the form of histogram. | |
| **Pre-Condition** | Image should be present in the system. | |
| **Post-Condition** | User can view histogram of image grey-levels. | |
| **Scenario** | **User Action** | **System Response** |
|  | Users will select the hist option on the screen. | The system will create histogram of the image upload by the user and will display it on the screen. |
| **Actor** | User | |
| **User Input** | User will click on the hist option on the screen of AIFS. | |
| **Expected Result** | Histogram will be displayed on the screen. | |
| **System Output** | Histogram displayed on the screen. | |
| **Status** | Pass | |
| **Created By** | Humna Hayee & Raheel Kiany | |
| **Executed By** | Humna Hayee & Raheel Kiany | |
| **Date** | 1st May, 2022 | |

|  |  |  |
| --- | --- | --- |
| **TEST CASE ID** | **TC:1.4** | |
| **Test Case Name** | Forgery Detection | |
| **Purpose** | To check forgery in the image user will apply method. | |
| **Pre-Condition** | User has uploaded image in AIFS. | |
| **Post-Condition** | Result of the user selected method will be displayed. | |
| **Scenario** | **User Action** | **System Response** |
|  | User select the method. | The system responds by applying the method selected by the user. |
| **Actor** | User | |
| **User Input** | User will select method. | |
| **Expected Result** | The system will apply the forgery detection method on the image. | |
| **System Output** | The system applies the method. | |
| **Status** | Pass | |
| **Created By** | Humna Hayee & Raheel Kiany | |
| **Executed By** | Humna Hayee & Raheel Kiany | |
| **Date** | 1st May, 2022 | |

**Table 5.5:** Forgery Detection

|  |  |  |
| --- | --- | --- |
| **TEST CASE ID** | **TC:1.5** | |
| **Test Case Name** | Export Results | |
| **Purpose** | To check either the data extracted exports in the form of file. | |
| **Pre-Condition** | User has extracted image data in AIFS. | |
| **Post-Condition** | User must have extracted data required. | |
| **Scenario** | **User Action** | **System Response** |
|  | User select the option of data export. | The system responds with a file contains image which user can save of other purpose use. |
| **Actor** | User | |
| **User Input** | User select data export option. | |
| **Expected Result** | The system will provide a file to the user. | |
| **System Output** | The system generated file with required data | |
| **Status** | Pass | |
| **Created By** | Humna Hayee & Raheel Kiany | |
| **Executed By** | Bilal Fawad Khan & Raheel Kiany | |
| **Date** | 4th July, 2022 | |

**Table 5.6:** Export Results

* + 1. **GUI Test Cases**

GUI test cases are done with some changes in the GUI and check the result for any inconsistency in the GUI when changed.

**Table 5.7:** Iniating AIFS GUI

|  |  |  |  |
| --- | --- | --- | --- |
| **Test** | **Description** | **Comment** | **Decision** |
| **Test description** | Initiating AIFS | To ensure that the application is installed in the system and working. | Pass |
| **Initial conditions** | Platform available to install the application. | Ensure everything works fine. |
| **Tests inputs** | The user click on the icon to open the application . | Window opens on screen. |
| **Expected results and criteria** | User opens AIFS successfully. | Recommended action is performed and user has successfully access to AIFS. |
| **Test outputs** | User has AIFS installed in system. | User can now upload image and use the AIFS system. |

**Table 5.8:** Image Upload GUI

|  |  |  |  |
| --- | --- | --- | --- |
| **Test** | **Description** | **Comment** | **Decision** |
| **Test description** | Image upload | To ensure that the code responsible for uploading the image. | Pass |
| **Initial conditions** | User has AIFS open. | Ensure everything works fine. |
| **Tests inputs** | Admin clicks on upload image. | Uses should select image file type is which is supported. |
| **Expected results and criteria** | The user successfully  Uploads image. | Recommended action is performed and image is successfully added. |
| **Test outputs** | Image uploaded by the user. | N/A |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test** | **Description** | **Comment** | **Decision** |
| **Test description** | Metadata extraction | To ensure that the code is responsible and extract meta-data of image. | Pass |
| **Initial conditions** | Image must exist in AIFS. | Ensure image is in the system and everything works fine. |
| **Tests inputs** | User clicks on MT button. | N/A |
| **Expected results and criteria** | The system successfully displays metadata. | User successfully get the detail requested from the system. |
| **Test outputs** | System successfully displays meta-data of the image. | Metadata extracted. |

**Table 5.9:** Metadata Extraction GUI

|  |  |  |  |
| --- | --- | --- | --- |
| **Test** | **Description** | **Comment** | **Decision** |
| **Test description** | Histogram display | To ensure that the code is responsive for creating histogram of image grey-levels. | Pass |
| **Initial conditions** | User must have the image in the AIFS. | Ensure everything works fine. |
| **Tests inputs** | User click on hist button | N/A |
| **Expected results and criteria** | The system displays the histogram to the user | User successfully gets the result from system |
| **Test outputs** | The system successfully displays histogram to the user. | User can view histogram. |

**Table 5.10:** Histogram Display GUI

* + 1. **Useability Testing**

Usability testing is the next step after implementation. User interfaces were tested multiple times to check whether the interfaces are easy to use or not. All the individuals who tried using the system were able to understand it´s working easily and results turned out to be successful.

* + 1. **Software Performance Testing**

When each module is complete and ready for testing, performance testing for the system will be done. Firstly, the performance of each module will be done separately then in collaboration with other modules, and the performance of the overall system will be evaluated based on the time needed to get the results and display it.

**Table 5.11:** Software Performance Testing

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step No.** | **Test Cases** | **Test Data** | **Expected Result** | **Actual Result** | **Status (Pass/Fail)** |
| 1 | Image loading speed | Operating system application | Image will load efficiently | Image loaded as expected | Pass |
| 2 | Making data file for export | Creating File | Creating file contains image data efficiently | System generated required file | Pass |

* + 1. **Compatibility Testing**

Compatibility testing is non-functional testing conducted on the application to evaluate the application's compatibility within different environments.

* Operating system Compatibility Testing - Windows 8 and Windows 10

**Table 5.12:** Compatibility Testing

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step No** | **Test Cases** | **Test Data** | **Result** | **Status**  **(Pass/Fail)** |
| 1 | Check that the software is compatible with the OS | Application | The OS versions are compatible with AIFS | Pass |

* + 1. **Installation Testing**

Installation testing is to determine whether the software application successfully installed.

**Table 5.13:** Installation Testing

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step No** | **Test Cases** | **Test data** | **Result** | **Status (Pass/Fail)** |
| 1 | Check that the software works on Windows 8 and higher versions. | Application | This software is working as intended | Pass |

* + 1. **Acceptance Testing**

Acceptance testing is performed to check the end user’s acceptance criteria. It is performed right before making the system available for actual use. It includes alpha testing and beta testing.

**Table 5.14:** Acceptance Testing – Alpha

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step No** | **Test Cases** | **Test data** | **Result** | **Status**  **(Pass/Fail)** |
| 1 | Check the spellings of words used in GUI of the AIFS application. | AIFS windows | Software does not contain spelling mistakes. | Pass |
| 2 | Check the response time required to access different functionalities | Application | Response time of different functionalities is minimum as expected. | Pass |

**Table 5.15:** Installation Testing – Beta

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step No** | **Test Cases** | **Test data** | **Result** | **Status (Pass/Fail)** |
| 1 | Check the response time required to access different functionalities | Application | Response time of different functionalities is minimum as expected. | Pass |

* 1. **Defect Report**

The defect is defined as a deviation from the expected and actual result of the application. It is also defined as the irregularity from the specification mentioned in the product functional specification document. Defects solved by the developer in the development phase.

**Table 5.16:** Defect Report – Image Upload

|  |  |
| --- | --- |
| **Bug ID** | 1.0 |
| **Test Case ID** | TC 1.1 |
| **Purpose** | Upload Image |
| **Expected Result** | Image is uploaded |
| **Actual Result** | Unsupportable image |
| **Bug Description** | Unsupportable image |
| **Testing Status** | Close |
| **Author** | Humna Hayee |
| **Tester** | Humna Hayee |
| **Severity** | Low |
| **Priority** | Low |

**Table 5.17:** Defect Report – Metadata Extraction

|  |  |
| --- | --- |
| **Bug ID** | 1.1 |
| **Test Case ID** | TC 1.3 |
| **Purpose** | Extracting meta-data of image |
| **Expected Result** | Meta-data of image should be extracted from the image and should be displayed to the user. |
| **Actual Result** | Image was not found in the system to extract meta-data. |
| **Bug Description** | Pre-condition was not full filled. |
| **Testing Status** | Close |
| **Author** | Humna Hayee |
| **Tester** | Humna Hayee |
| **Severity** | Medium |
| **Priority** | Low |

* 1. **Test Report**

This section includes the report of all the tests that were performed or chosen to be performed. All of them were executed throughout the project and reported back in this table. This table is according to the last versions of test cases; hence all the test cases are passed. It also includes pending, in progress, blocked, or deferred test cases.

**Table 5.18:** Test Report

|  |  |
| --- | --- |
| Testing Activity | Count |
| Tests Passed | 19 |
| Test Failed | 0 |
| Total Executed | 19 |
| Pending | 0 |

Chapter 6

# CONCLUSION

* 1. **Introduction**

This chapter concludes our efforts and project milestones, as it is the final portion of the documentation. It enlightens all of the hard work put in during the project on what we were able to achieve. It contains details about the project's goal and its constraints.

* 1. **Conclusion**

The framework makes it easy for users to upload an image and request the system to check if there is any forgery in the image. This method is explicitly designed to detect forgery using minimal time-consuming mechanism in which users often find themselves get result quickly. Since time is so valuable, this framework will concentrate on providing the consumer with the results of their requests in a timely and efficient manner. Moreover, the users can find relatable information of the image in the form if metadata and histogram.

* 1. **Benefits**

This project has made it easier for people to find the difference between real and imprinted image. Detecting image forgery and extracting metadata for it helps the user to understand more detail about the image. The application also allows users to make histogram of image which can display image grey level this information can help to refine image.

* 1. **Future Work**

In the future, Alhazen Image Forensics System will offer more methods to detect forgery in the image based on latest research papers. Through integration of these features, we will be able to provide better result. It is also in our plans to convert it into video editor in which we will divide video into small frames, on which different methods will be applied.

# Bibliography

|  |  |
| --- | --- |
| [1] | J. Huang, W. L. and H. L. , “Localization of Diffusion-based Inpainting in,” *IEEE Transactions on information and Security,* p. 15, 27 July 2017. |
| [2] | Sural, Pramanik and G. Qian, “Segmentation and histogram generation using the HSV colour space for image retrieval,” in *IEEE International Conference on Ubi-Media Computing*, Lanzhou, China , 2008. |
| [3] | Abhishekchauhan16, “GeeskForGeeks,” 2021 December 2021. [Online]. Available: https://www.geeksforgeeks.org/multithreading-python-set-1/. [Accessed 22 03 2022]. |