

**Document Version 1.0**

**IriEnvoy Desktop App Testplan**

**Document Change Record**

This page records any updates and revisions made to the IrisCapture+ Test plan.

|  |  |  |  |
| --- | --- | --- | --- |
| **Doc ver.** | **Date** | **Change Description** | **Software ver.** |
| 1.0 | 2024 Oct22 | iricapture+.aar |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Table of Content**

[1. Introduction 4](#_heading=h.gjdgxs)

[1.1. Objectives(Mục tiêu) 4](#_heading=h.30j0zll)

[1.2. Testing Strategy(Chiến lươc kiểm thử) 5](#_heading=h.1fob9te)

[1.3. Scope(Phạm Vi) 5](#_heading=h.lnxbz9)

[1.4. Related Documents 5](#_heading=h.35nkun2)

[1.5. Definitions and Acronyms 5](#_heading=h.1ksv4uv)

[1.6. Schedule and Milestones and Deliverables 5](#_heading=h.44sinio)

[2. Resource Requirements 6](#_heading=h.z337ya)

[2.1. Hardware 6](#_heading=h.3j2qqm3)

[2.2. Software 6](#_heading=h.4i7ojhp)

[2.3. Staffing 7](#_heading=h.1ci93xb)

[3. Stage of Test 7](#_heading=h.qsh70q)

[4. Testing standard 7](#_heading=h.3as4poj)

[5. Features To Be Tested 7](#_heading=h.1pxezwc)

[5.1. New Features Testing 7](#_heading=h.49x2ik5)

[5.2. Regression Testing 7](#_heading=h.2p2csry)

[6. Features Not To Be Tested 7](#_heading=h.147n2zr)

[7. Dependencies/Risks 7](#_heading=h.3o7alnk)

[8. Entrance/Exit Criteria 7](#_heading=h.23ckvvd)

# Introduction

## Objectives (Mục tiêu)

This document describes the testing methodology and procedures for IriEnvoy-MK, including hardware and software. The objective of these testing activities is to ensure that the functions operate according to the specifications and documentation.

## Testing Strategy (Chiến lươc kiểm thử)

## Strategy to build test case:

## Test the device's USB connectivity to the computer

## All Frontend and Backend functions must write test cases

## Test device functions such as capture, storage, ...

## Test device

## Focus testing on Windows 10, 11

## Then, test hardware such as checking USB 2.0 ports, IriEnvoy-MK device.

## Resource strategy:

## Test: Responsible for testing Functions, Interfaces, and Devices

## Review the results and discuss with the developer

## Test priority order:

## Prioritize testing of photo capture and storage functions.

## Interface: Execute Manual test

## Performance Testing: Measures device processing time and recognition speed

## Scope (Phạm Vi)

1. Functional requirements for test execution:

* Frontend: Interface testing
* Backend: Data processing capabilities including capture, storage of results

2. Test execution environment:

* Device: IriEnvoy-MK camera unit, Windows

## Related Documents

https://drive.google.com/drive/folders/1BU-UcUi6JhdiPCqVCdB5CShEbpfxaUCr

## Definitions and Acronyms

## Schedule and Milestones and Deliverables

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No | Milestone Task | Effort (Hour) | Assign to | Deliverables | Delivery date | Note |
| 1 | Create Test Plan | 4 | Phước, Phượng | IriEnvoyTestplan.docx | 21/10/2024 |  |
| 2 | Create Test case File  (Test Functure ) |  | Phước, Phượng | IriEnvoyTestplan. xlsx |  |  |
| 3 | Make test report |  | Phước, Phượng |  |  |  |
| 4 | Create Test case File  (Interface Testing) |  | Phước, Phượng |  |  |  |

# Resource Requirements

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Name | Rule | Task |
| 1 | Phước, Phượng | Tester | - Analyzing project requirements, synthesizing related documents  - Work planning  - Create Unit Android Test case and execute test  - Make test report |
| 2 | Phước, Phượng | Tester | - Test the interface as required  - Test the function as required  - Create Test cases (Manual Test UI)  - Make test report |

## Hardware

* IriEnvoy-MK camera unit

## 2.2 Software

* IriEnvoy-MK\_Iris\_captureDemo.msi: UI testing

### Test Tool

* Desktop application testing: Visual Studio
* Test Plan: DOC
* Test Case: MS Excel
* Result Test: MS Excel

## Staffing

* Tester

### Responsibilities

### Training

# Stage of Test

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test Types | Unit test | Integration | System | Acceptance |
| Interface Test | x |  | x | x |
| Functional Test | x |  | x | x |
| Load test |  |  | x | x |
| Usability test |  |  | x | x |

# Testing standard

|  |  |
| --- | --- |
| Start condition test | - Round 1: When the developer has completed 70% of the modules, Unit testing has been done  - Set up environment, configure for testing  - Round 2: Test 30% of the remaining modules, and retest the bugs in round 1  - Round 3: Retest the bugs in Round 2 |
| When stop test | - Complete test  - All bugs must be credited to an agreed solution. No more bugs with Medium level or higher |
| Standard success test | - Satisfy the functional requirements of the software  - Satisfying user requirements |
| When does test recurrent | - Get new libs and new app from developer  - At the request of developer |

# Features To Be Tested

## New Features Testing

* Device settings
* Iris image capturing
  + Auto capture mode
  + Operator capture mode
* Capture settings
  + Capture mode (Time-based vs Frame-based)
  + Streaming option
  + Streaming image format
  + Result image option
    - Image kind
    - Image format
* Image quality
  + Assess captured image quality (sharpness, contrast, iris pattern visibility)
  + Test with and without glasses/contact lenses
* Error Handling
  + Verify appropriate error messages for common issues (e.g., no qualified frame)

## Regression Testing

* Existing Feature Verification: Re-run all test cases from sections 3.1 to 3.6 to ensure existing functionality hasn't been affected by the new features
* Integration Testing:
  + Verify that the new features integrate smoothly with existing functionality
  + Test workflows that combine new and existing features
* Performance Regression
  + Compare system performance metrics (CPU usage, memory consumption, response times) before and after the introduction of new features
  + Ensure that the addition of new features hasn't negatively impacted overall system performance
* UI/UX Regression
  + Verify that the addition of new features hasn't adversely affected the user interface layout or responsiveness
  + Check for any unintended changes in UI element.
* Compatibility Regression
  + Re-test compatibility with all supported operating systems and hardware configurations
  + Verify that the new features haven't introduced any new compatibility issues
* Error Handling Regression
  + Re-test all error scenarios to ensure that error handling for existing features still functions correctly
  + Verify that new features haven't introduced any new uncaught exceptions or error states

## Software installation test

* Installation Process
* Driver Installation: Confirm automatic installation of Microsoft's USB Video Class (UVC) driver

## Performance test

* Capture Speed
* Resource Usage

# Features Not To Be Tested

## Mobile applications

## Hardware components of the iris scanning device

# Dependencies/Risks

## Dependencies

* IriEnvoy-MK
* Goggle
* Software package

## Risks

* Availability of test devices and environments

# Entrance/Exit Criteria

## Entrance Criteria

* Required test devices are available and in working condition
* Test environment is set up and configured

## Exit Criteria

* All new features have passed acceptance criteria
* All planned test cases have been executed
* No critical or high-priority bugs are open

**- The end of the document -**