

Scenario 5: Module Development & Deployment for Developer

Mission :

Identify a missing feature or improvement opportunity through personal use or community discussion.

Create a new module using the Luxoria.Modules, Luxoria.GModules & Luxoria.SDK.

Use CLI tool (buildsystem.py) to build the module locally on the developer's workstation.

Upload the module to the Luxoria Marketplace using the API or developer portal.

Provide metadata (version, changelog, dependencies).

Participate in GitHub discussions and propose changes or fixes via pull requests.

Join the official Discord for community interaction and feedback loops.

Update the module based on feedback and release a patch version.

Expected Result :

Developer successfully builds a module using official Template, Luxoria Standard Modules, SDK and CLI tool (LuxCLI).

Marketplace accepts the module upload with correct metadata.

Community interaction channels (GitHub, Discord) support collaboration and feedback

Bug fixes and feature updates are released incrementally and traceably

Module functions reliably within the Luxoria ecosystem, following modular design guidelines.

1. Module Creation

The module which will be created, will be called “LuxImport3”, it will be based on the actual version of LuxImport.

1.1 Fork <https://github.com/LuxoriaSoft/marketplace/> repository

Create a new fork

A fork is a copy of a repository. Forking a repository allows you to freely experiment with changes without affecting the original project.

Required fields are marked with an asterisk (*).

Owner * Repository name *

 qerdinger / marketplace

✔ marketplace is available.

By default, forks are named the same as their upstream repository. You can customize the name to distinguish it further.

Description (optional)

Luxoria Marketplace API / Listings

☐ Copy the `main` branch only
Contribute back to LuxoriaSoft/marketplace by adding your own branch. [Learn more.](#)

 You are creating a fork in your personal account.

Create fork

1.2 Clone the forked repository

```
[→ git clone https://github.com/qerdinger/marketplace.git
Cloning into 'marketplace'...
remote: Enumerating objects: 1082, done.
remote: Counting objects: 100% (14/14), done.
remote: Compressing objects: 100% (13/13), done.
remote: Total 1082 (delta 5), reused 2 (delta 1), pack-reused 1068 (from 1)
Receiving objects: 100% (1082/1082), 83.38 MiB | 42.65 MiB/s, done.
Resolving deltas: 100% (434/434), done.
```

1.3 Select the targeted version of Luxoria

1.4 Create LuxImport3

```
{
  "luxmodversion": 1,
  "name": "LuxImport3",
  "description": "The thrid version of LuxImport",
  "author": "The Luxoria Team",
  "email": "contact@luxoria.blupelicansoft.com",
  "license": "Apache-2.0",
  "url": "https://docs.luxoria.blupelicansoft.com/modules/luximport3",
  "repository": "https://github.com/luxoriasoft/luxoria",
  "compatibility": {
    "mininumVersion": ">=1.0.0",
    "maximumVersion": "<=99999.0.0"
  },
  "dependencies": {
    "LuxCore": ">=2.0.0"
  },
  "keywords": [
    "Lux",
    "Import",
    "Standard Importation Module",
    "StdMod",
    "Standard",
    "Base",
    "Fondation"
  ],
  "brochure": "README.md",
  "build": {
    "config": "Release",
    "csproj": "LuxImport\\LuxImport.csproj",
    "dll": "LuxImport.dll",
    "bin": "LuxImport\\bin",
    "runtimes": {
      "x64": "win-x64",
      "x86": "win-x86",
      "arm64": "win-arm64"
    },
    "targetFramework": "net9.0-windows10.0.26100.0"
  }
}
```

LuxImport3

This is the third version of LuxImport
Which improves UI + Logics

LuxImport is a image collection management system designed for efficient indexing, asset organization, and metadata handling. It provides capabilities for importing, processing, and managing image datasets, making it an ideal solution for large-scale photography and media applications.

Architecture

LuxImport follows a modular architecture to ensure scalability, maintainability, and high performance. The key components are:

1. Core Components

- **ImportService**: Handles the import process, including asset hashing, metadata extraction, and database initialization.
- **Manifest Repository**: Manages the manifest file that keeps track of indexed assets.
- **LuxConfig Repository**: Stores configuration settings related to collections and assets.
- **FileHasherService**: Computes unique hashes for images to prevent duplication.
- **SkiaSharp Integration**: Used for image processing, including metadata extraction and transformations.

2. Data Flow

1. **Initialization**: The system checks whether the target collection exists and initializes it if necessary.
2. **Indexing**: Images are scanned, hashed, and added to the database with relevant metadata.
3. **Storage & Retrieval**: Indexed assets are stored with reference to their respective manifest files.
4. **Loading Assets**: Assets are loaded from storage and processed based on application requirements.

3. Technologies Used

- **.NET 9**: Primary Framework.
- **BenchmarkDotNet**: Used for performance benchmarking.
- **System.Threading.Tasks**: Ensures efficient async processing.
- **Segmented Collections**: Optimizes parallel data processing.

1.5 Commit + Push LuxImport3

```
→ git push
Enumerating objects: 67, done.
Counting objects: 100% (67/67), done.
Delta compression using up to 10 threads
Compressing objects: 100% (50/50), done.
Writing objects: 100% (65/65), 108.54 KiB | 108.54 MiB/s, done.
Total 65 (delta 14), reused 52 (delta 11), pack-reused 0
remote: Resolving deltas: 100% (14/14), completed with 2 local objects.
To https://github.com/qerdinger/marketplace.git
f818501..a530902 v1.56.0 -> v1.56.0
```

2 Create a pull request to Marketplace

“To integrate LuxImport3 to marketplace, we need to create a pull request that target the targeted version such as LuxoriaSoft:v1.56.0”

[+ Luxoria v1.56.0] LuxImport @ 3.0.0 #2

 Open qerdinger wants to merge 1 commit into [LuxoriaSoft:v1.56.0](#) from [qerdinger:v1.56.0](#) 


Conversation 0

Commits 1

Checks 0

Files changed 49

+18,391 -0

 qerdinger commented now Member

LuxImport @ 3.0.0

Description

New LuxImport v3

- New UI
- New Logic
- Better performance

Developpement

.NET 9.0 has been used, such as SkiaSharp 3.116.0

Information

This is the new version of LuxImport Version 2.0

Documentation

Documentation at : <https://docs.luxoria.bluepelicansoft.com/modules/luximport>

Reviewers

Lucas25000

epi-noahg

Still in progress? [Convert to draft](#)

Assignees

qerdinger

Labels

enhancement

Projects

None yet

Milestone

No milestone

Development

Successfully merging this pull request may close these issues.

None yet

Notifications

Customize

2.1 Wait until the PR has been reviewed & accepted

“A reviewer/administrator has reviewed the PR and accepted it”

Check it out at:

<https://github.com/LuxoriaSoft/marketplace/pull/2>

+18,391 -0 ■■■■■■

Reviewers



epi-noahg



Lucas25000



Still in progress? [Convert to draft](#)

2.2 The PR has been merged



Pull request successfully merged and closed


You're all set — the `v1.56.0` branch can be safely deleted. If you wish, you can also delete this fork of LuxoriaSoft/marketplace in the [settings](#).

Delete branch

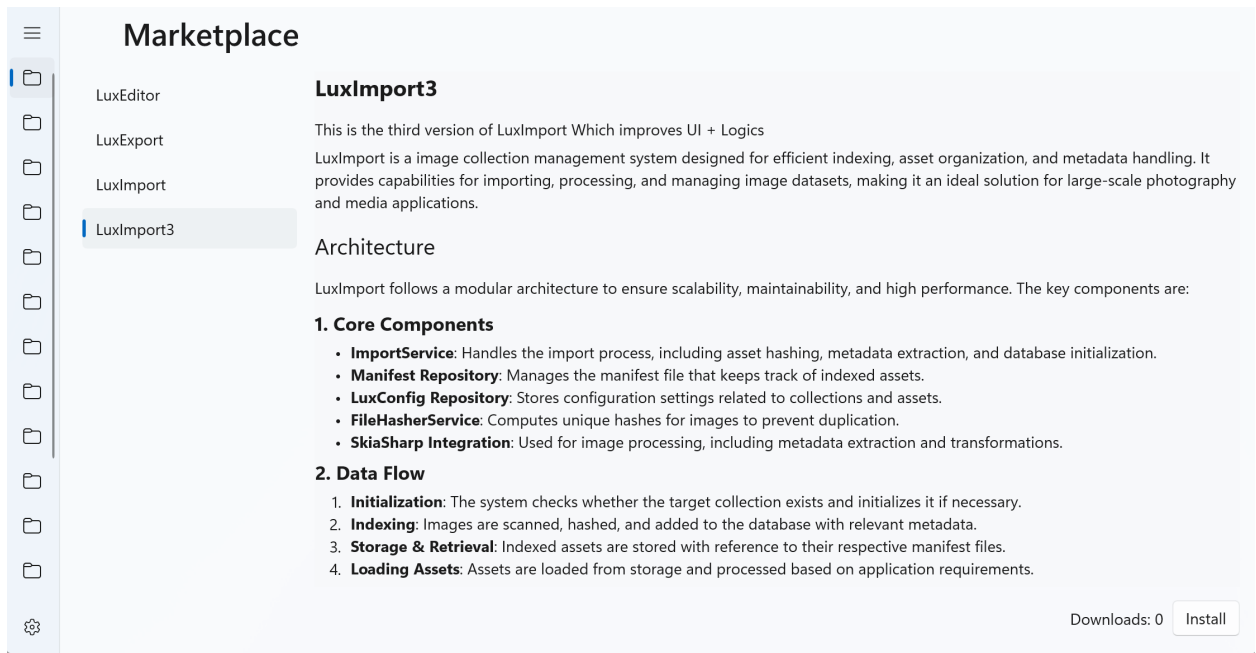
The module will be soon available on the Marketplace (Estimated: 5-10 minutes)

2.2 Keep posted/updated the community on “Luxoria Community”

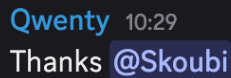
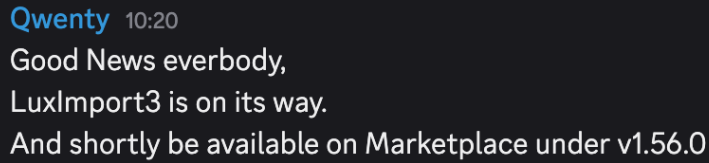
5 July 2025

**Qwenty** 10:20
Good News everybody,
LuxImport3 is on its way.
And shortly be available on Marketplace under v1.56.0

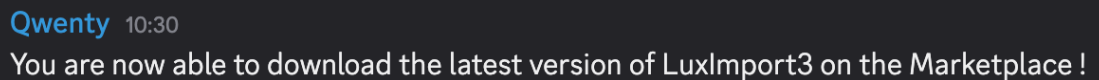
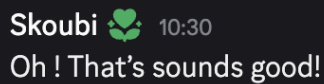
2.3 Be the first to install the surprisingly new LuxImport3



3 Chat with “Luxoria Community”



This has been the new version of LuxImport3
Which contains a lot of improvements compared to the original LuxImport



LuxEditor

LuxExport

Luximport

LuxImport3

LuxImport3

This is the third version of LuxImport Which improves UI + Logics

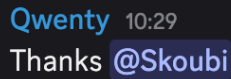
LuxImport is a image collection management system designed for efficient indexing, asset organization, and metadata handling. It provides capabilities for importing, processing, and managing image datasets, making it an ideal solution for large-scale photography and media applications.

Architecture

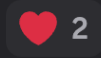
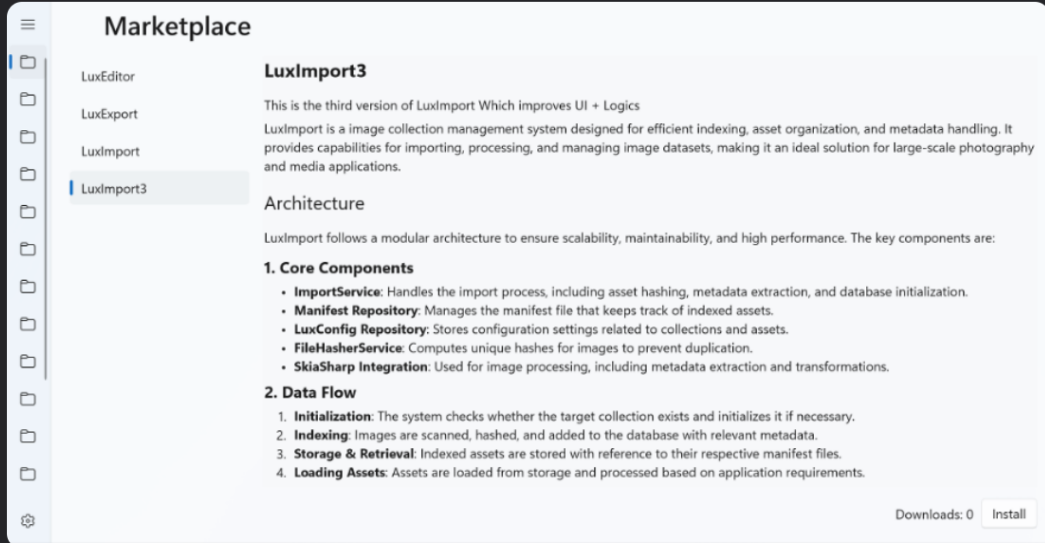
LuxImport follows a modular architecture to ensure scalability, maintainability, and high performance. The key components are:


1. Core Components

- **ImportService:** Handles the import process, including asset hashing, metadata extraction, and database initialization.
- **Manifest Repository:** Manages the manifest file that keeps track of indexed assets.
- **LuxConfig Repository:** Stores configuration settings related to collections and assets.
- **FileHasherService:** Computes unique hashes for images to prevent duplication.
- **SkiaSharp Integration:** Used for image processing, including metadata extraction and transformations.



You are now able to download the latest version of LuxImport3 on the Marketplace !



Skoubi  10:31
I will give it a try !