BETA TEST PLAN – LUXORIA

1. Core Functionalities for Beta Version

Luxoria is composed of several components, each with its own set of features. Below is a summary of the core functionalities that will be included in the beta version.

Luxoria Desktop App

The Luxoria Desktop App is the main application that will be used by photographers. It will include features for collection importation, filtering, editing, and exportation, while communicating with the Luxoria Studio for real-time collaboration and feedback.

Feature Name	Description	Priority (High/Medium/Low)	Changes Since Tech3
Luxoria.App (Windows-based application)	Core of the application, mainframe providing handlers for system updates, UI management	HIGH	Technology switches from .NET WPF to .NET WinUI3, Spin-off of the application from monolitic app to event-driven architecture
Luxoria Modular Graphical User Interface (LMGUI)	Embedded graphical engine for designing consistent, modular interfaces within application modules	HIGH	Creation of a new way to design modular interfaces from modules, used to create graphical interfaces
Luxoria Universal/External Modules (Luxoria.Modules)	Handles the lifecycle of modules (load, manage, communication between them and the application's core), enables extensibility of core features	HIGH	A sister component, initially thought in Luxoria itself
Luxoria Universal/External UI Modules (Luxoria.GModules)	User interface layer that connects external modules to LMGUI in a streamlined and consistent way	HIGH	Overlay based on Luxoria.Modules, brings an UI implementation
Luxoria Software Development Kit (Luxoria.SDK)	Provides internal logging tools and communication with Sentry for crash reporting and diagnostics	HIGH	Initially none, this part is throught for open-source & collaboration

Feature Name	Description	Priority (High/Medium/Low)	Changes Since Tech3
Importation (LuxImport)	Imports folders from the local system to initialize a collection of images	MEDIUM	-
Filtering (LuxFilter)	Uses AI-based algorithms (quality, sharpness, U-NET segmentation) to sort and assess images	LOW	-
Editing (LuxEditor)	Allows image editing via color sliders, layer controls, and preset applications	MEDIUM	Initially CPU-based, we've selected an GPU-based approach
Exportation (LuxExport)	Exports images in various formats with options for file naming, visual quality, and watermarking	MEDIUM	-
LuxStudio Integration (LuxStudioInt)	Integrates customer feedbacks & picture status, real-time chat integrations, upload files to LuxStudio, etc.	LOW	A sister component, initially thought in Luxoria itself, Real-Time communcation using WebSocket (.NET SignalR) / An event management (Queue) using RabbitMQ

LuxStudio

Luxoria is a collaborative platform for photographers and clients, allowing for real-time communication and image management. The Luxoria Studio component will be the main interface for users to interact with collections, images, and each other.

Feature Name	Description	Priority (High/Medium/Low)	Changes Since Tech3
SSO Authentication	Authentication via third-party providers for easy access.	High	Not in Tech3. Added to improve UX and unify login system across the platform.
Role-Based Access Control	Different access levels (Owner, User, Guest) with permission enforcement.	High	Fully implemented in frontend/backend using JWT and external authentification (SSO).
Collection Sharing	Share collections securely with specific users via email invitation.	High	Added invitation via email + enforced through authenticated identity (SSO login).

Feature Name	Description	Priority (High/Medium/Low)	Changes Since Tech3
Image Upload & Viewer	Upload images to a collection and view them in a responsive gallery.	High	Integration with MinIO instead of raw file system. Image paths and storage fully externalized
Bulk Upload	Upload multiple images at once via drag-and-drop interface.	Medium	Not initially planned – added due to UX needs and user interviews.
Image Download	Ability to download single or multiple images from a collection.	Medium	Not specified in Tech3 – added after feedback from photographers needing quick export.
Feedback System	Comment on specific images for review or validation.	Medium	Moved from web-only vision to shared persistence in both frontend and DB for feedback loop.
Real-Time Chat	Live messaging between collaborators on each collection.	High	Originally static comments – upgraded to SignalR-powered persistent group chat.
Mentions in Chat	Mention users or reference images using @ syntax inside messages.	Low	New feature idea added after first prototype testing – enhances chat interactivity.
Activity Log	Track user actions (uploads, comments, deletions) within a collection.	Medium	Introduced to improve traceability; not in original scope. Initial backend implemented.

Luxoria Website

This is the public-facing website for Luxoria, which will serve as a marketing and information hub. It will also include a marketplace for modules.

Feature Name	Description	Priority (High/Medium/Low)	Changes Since Tech3
Testimonials & User Experiences	Showcase user reviews and testimonials to build credibility	Low	-
GitHub Repository Link	Provide a direct link to the project's GitHub repository	Medium	-
Github Insights	Display project metrics (Downloads) fetched from GitHub API	Low	-

Feature Name	Description	Priority (High/Medium/Low)	Changes Since Tech3
Server-Side Rendering (SSR)	Next.js SSR to improve SEO and initial-load performance	High	-

Luxoria Marketplace API

This part is not directly related to the Luxoria application but is essential for the Luxoria Marketplace. It will be used to manage modules and their versions, allowing developers to upload and download modules programmatically.

Feature Name	Description	Priority (High/Medium/Low)	Changes Since Tech3
Download latest Luxoria Version	Provides access to the latest stable version of Luxoria for automated updates	High	Not initially thought
Upload Modules	Allows developers to upload their own modules to the marketplace	Medium	Not initially thought
Download Modules	Enables clients to retrieve modules programmatically from the marketplace	High	Not initially thought
Metadata For Releases	Offers metadata (version, changelog, dependencies) for each module or release	High	Not initially thought
Programmatic Endpoints	Provides structured API endpoints for automation and integration with CI/CD	High	Not initially thought

2. Beta Testing Scenarios

2.1 User Roles

Luxoria will be tested by various user roles, each with different needs and expectations. Below is a summary of the roles and their descriptions.

Role Name	Description
Lambda (Casual User / Curious Visitor)	Non-technical, explores out of curiosity. Might download Luxoria to try it out, Simplicity, visual feedback, basic filtering, and easy exports.

Role Name	Description
Amateur Photographer (Enthusiastic Hobbyist)	Loves photography, has a decent camera, edits images regularly but not professionally, Assistance with quality evaluation, intuitive editing tools, ability to share results.
Professional Photographer (Working Photographer)	Sells or delivers images to clients, needs quality and efficiency, Batch operations, advanced control, reliable collaboration, professional export tools.
Luxoria Administrator (Internal or Power User)	Maintains collections, manages LuxStudio teams, supports other users, Control over user access, collection status, sync/logs, and module management.
Developer (Technical User)	Wants to extend Luxoria, build modules, or script image processing, APIs, module system, clear documentation, debugging tools.

2.2 Test Scenarios

Scenario 1: Basic Photo Workflow

- **Role Involved:** Lambda (Casual User / Curious Visitor)

- **Objective:** Ensure a casual user can perform basic photo editing and export with minimal technical knowledge.

- **Preconditions:**

    - Luxoria Desktop is installed.
    - A folder with sample images is available on the user's computer.

- **Test Steps:**

    1. Launch Luxoria Desktop.
    2. Click on "Import" and select a folder with images.
    3. Skip the filtering process.
    4. Open a selected image in the editor.
    5. Adjust the exposure using the slider.
    6. Crop the image using the cropping tool.
    7. Click on "Export" choose JPEG format, and set quality to low.
    8. Save the image to a local folder.

- **Expected Outcome:**

    - The folder is imported with one click.
    - User can easily edit exposure and crop the image.

- Export works correctly with the selected format and reduced quality.
- The interface remains visually intuitive and accessible throughout the process.

Scenario 2: Basic Photo Workflow for Enthusiastic Hobbyist

- **Role Involved:** Amateur Photographer (Enthusiastic Hobbyist)

- **Objective:** Verify that an amateur photographer can import RAW photos, quickly sort and enhance them with AI filters and presets, then export at full quality.

- **Preconditions:**

    - Luxoria Desktop is installed and open.
    - A local folder contains image files.

- **Test Steps**

    1. Launch Luxoria Desktop.
    2. Click on "Import" and select a folder with images.
    3. Apply the built-in AI filters to automatically flag the best shots, manually deselect any remaining images as needed.
    4. Open a flagged image in the Editor
    5. Apply a built-in preset
    6. Tweak the colorimetry sliders (e.g. exposure, temperature).
    7. Use the Layers tool to detach the subject from the background.
    8. Apply a blur effect to the background layer.
    9. Select all edited images, click Export
    10. Choose JPEG at 100% quality, set a naming template, and export to the chosen folder.

- **Expected Outcome:**

    - All RAW files appear as thumbnails after import.
    - AI filtering correctly highlights top images for review.
    - Preset application and manual color adjustments update the image in real time.
    - The Layers tool cleanly isolates the subject and blurs the background.
    - Exported JPEGs at full quality are named per template and saved without errors.

Scenario 3: Professional Workflow with Collaboration for Photo. Professionnel

- **Role Involved:** Professional Photographer (Working Photographer)

- **Objective:** Ensure a professional user can perform high-quality batch edits, create & use presets, apply watermarking, and collaborate efficiently with clients via LuxStudio.

- **Preconditions:**

- Luxoria Desktop and LuxStudio accounts are set up.
- A folder of RAW images is available.
- Client feedback system is enabled in LuxStudio.

- **Test Steps:**

  1. Launch Luxoria Desktop
  2. Import a folder of RAW files using the RAW reader.
  3. Run the filtering process to score and sort images by quality and sharpness.
  4. Select a subset of top-rated images for editing.
  5. Apply advanced edits using layer tools and color calibration features.
  6. Create and save a new preset based on these edits.
  7. Link the collection to a LuxStudio project and set visibility to collaborators.
  8. Export the edited images with a watermark in high-quality JPEG format on the online collection.
  9. Invite clients or colleagues for review via LuxStudio.
  10. Receive feedback through the embedded chat, apply requested changes, and update the images.
  11. Mark the collection as "Final" and share the link with guests for delivery.

- **Expected Outcome:**

  - RAW files are correctly read and imported.
  - Filtering ranks images accurately by visual quality.
  - Editing tools (layers, calibration) function professionally.
  - Presets can be saved and reused.
  - Export includes watermarked versions with correct naming and formats.
  - Collaboration via LuxStudio is seamless: feedback is visible, changes applied, and final assets shared efficiently.

Scenario 4: Admin Workflow – Managing Collections and Access

- **Role Involved:** Luxoria Administrator

- **Objective:** Validate that the administrator can access administrative features such as user and role management, activity tracking, and module oversight via the Admin Center.

- **Preconditions:**

  - The administrator is authenticated.
  - At least one collection and one user exist in the system.
  - Admin permissions are enabled for the connected account.
  - Marketplace API is active.

- **Test Steps:**

1. Log in to LuxStudio using an admin account.
2. Navigate to the Admin Center from the main navigation.
3. Open the Access Control panel and view current users and their roles.
4. Add a new user by email and assign a role ("User", "Admin").
5. Save changes and confirm that the user is now listed with the correct role.
6. Navigate to the Activity Log and confirm that the role assignment is recorded.
7. Access the Reporting section and view a summary of all reports or abuses.
8. Ban a suspicious user.
9. Navigate to Marketplace API Management
10. View the list of the most downloaded/trending unapproved modules.
11. Accept or reject the new submission.
12. Connect to https://luxoria.sentry.io, to view the crash reports and crash analytics.
13. Access to the online feedback collection system (https://feedback.luxoria.bluepelicansoft.com) to view the feedbacks and suggestions from users.

- **Expected Outcome:**

    - The administrator successfully accesses the Admin Center and all its sub-sections.
    - The list of users and roles is correctly displayed, and new users can be added with valid roles.
    - Role assignment and banning actions are reflected immediately in the system and recorded in the Activity Log.
    - The Reporting section displays all flagged content or user reports accurately.
    - Suspicious users can be banned, and the action is logged.
    - The Marketplace API panel shows pending modules, trending downloads, and allows the admin to accept or reject submissions.
    - Sentry is accessible and displays up-to-date crash reports and analytics.
    - The external feedback platform loads correctly, showing user feedback and suggestions in real time.
    - System responds with no critical errors or performance issues throughout the workflow.

Scenario 5: Module Development & Deployment for Developer

- **Role Involved:** Developer / Technical User

- **Objective:** Validate that a developer can create, test, debug, and deploy a custom module for Luxoria via the SDK and Marketplace API.

- **Preconditions:**

    - Access to `docs.luxoria.bluepelicansoft.com` (public access).

- GitHub account and access to relevant repos (https://github.com/luxoriasoft/luxoria) (public access without permission).
- Registered account on the Luxoria Marketplace.

- **Test Steps:**

  1. Identify a missing feature or improvement opportunity through personal use or community discussion.
  2. Create a new module using the Luxoria.Modules, Luxoria.GModules & Luxoria.SDK.
  3. Use CLI tool (buildsystem.py) to build the module locally on the developper's workstation.
  4. Upload the module to the Luxoria Marketplace using the API or developer portal.
  5. Provide metadata (version, changelog, dependencies).
  6. Participate in GitHub discussions and propose changes or fixes via pull requests.
  7. Join the official Discord for community interaction and feedback loops.
  8. Update the module based on feedback and release a patch version.

- **Expected Outcome:**

  - 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.

3. Success Criteria

The following criteria will be used to determine the success of the beta version.

Luxoria Desktop App

Criterion	Description	Threshold for Success
Stability	No major crashes or critical bugs	Minor or Handled crashes could be accepted (Minor crashes on Sentry)
Usability	Users can navigate and understand features with minimal guidance	80% positive feedback from testers
Performance - Import	The import part (LuxImport) should be fluid	90% of positive feedback

Criterion	Description	Threshold for Success
Performance – Filter	The filtering part (LuxFilter) should be 10 times faster than human processing	90% of positive feedback
Performance – Editor	The editor part (LuxEditor) should be fast enough for fluidity	90% of positive feedback
Performance – Export	The export part (LuxExport) should process each asset in a resonable amount of time, the overall process depends on the number of assets, file size, format, quality, etc.	< 3sec for < 10MB / < 10sec for 10-100MB / < 30sec for 100MB+, the overall time should scale linearly but remain efficient, exporting 1,000 assets should complete in under 30 minutes
Accuracy – Import	The import part (LuxImport) should be able to accurately import assets without data loss	99% of positive feedback from testers
Accuracy – Filter	The filtering part (LuxFilter) Foreground/Background AI & algorithms should provide an overall score, representative of the real asset	60% accuracy from testing dataset OR 75% of positive feedback from testers
Accuracy – Editor	The editor part (LuxEditor) should provide accurate editing capabilities	90% of positive feedback from testers
Accuracy – Export	The export part (LuxExport) should export all assets without losing them (lossless) or crashing during processing	Less than 1% failure rate, ensuring robustness in handling different formats and qualities

LuxStudio

Criterion	Description	Threshold for Success
Stability	Must not crash or continue to function afterwards	Must not be offline for bugs or maintenances for more than 20 minutes at a time and 300 minutes in total (15-25 deployments/patches in total)
Usability	Users can navigate and understand features with minimal guidance	80% positive feedback from testers
Performance – Real-Time Features	Chat and feedback must sync instantly between users	95% of actions processed in =< 1 second
Performance – Account Verification System	The verification method for account creation should be instantly executed by our system	99% are traited under (<= 1) one minute
Accuracy – Notification System	Notification should be handled by our system and be executed without errors	99% notifications successfully processed

Criterion	Description	Threshold for Success
Accuracy - Access Control	Role-based permissions must be strictly enforced	100% of unauthorized actions blocked
Accuracy - Download Collection	Users must be able to export collections without error	98% of export actions complete successfully
Accuracy - Feedback Loop	Feedback from users must be visible to admins in the dashboard	100% of submitted feedback accessible in admin tools

Luxoria's Website

Criterion	Description	Threshold for Success
Stability	No crashes	Must not be offline during the beta
Usability	User should expect a fluid experience on our website	90% positive feedback from testers
Performance	Our website must be referenced by search engines.	Must be listed on Google

Luxoria Marketplace API

Criterion	Description	Threshold for Success
Stability	Must not crash or continue to function afterwards	Must not be offline for bugs or maintenances for more than 10 minutes at a time and 120 minutes in total (12 deployments/patches in total)
Usability	Operates & provides services as usual	95% of downloads of modules without any errors

4. Known Issues & Limitations

Issue	Description	Impact	Planned Fix? (Yes/No)
AI model (U-NET Image Segementation)	The model is currently underfitting under images who have not be trained on	Medium	No (Planned for post-beta fix)

Issue	Description	Impact	Planned Fix? (Yes/No)
RAW formats	The complete workflow and usage of all RAW formats are not entirely handled by our solution, when it comes to using all specifications or advanced treatments	High	No (Planned for post-beta fix)

5. Conclusion

Our solution is designed modularly, in many different parts such as Luxoria.App, each module, LuxStudio, LuxMarketplace.

To carry out our Beta Test Plan, we needs to validate each core functionality, module and part.

We will have several challanges to overcome, such as the data flow of many users uploading at the same time their collections and the general performance of the solution, both of Luxoria.App and LuxStudio.

By carrying out all of the above informations and details, we will be able to validate our solution and gather precious feedbacks through users, this will help our team to prepare for the final version of Luxoria.