

# ***Image Processing Project Guidelines***

## **Objective:**

The aim of this project is to apply image processing concepts to a chosen use case. You may select one of the suggested topics or propose an alternative, provided you clearly outline your project rationale and plan. Early iterations do not need to be perfect; what matters is that you demonstrate a logical thought process and progressively refine your solution based on testing and feedback.

## **Final Deliverables:**

- **Code Repository:** Submit your final code on GitHub with a clear commit history that reflects your project's progress.
- **Final Report:** A concise report summarizing your objectives, process, test outcomes, improvements, and final results.

## **Phase 1: Rationale (Weeks 2–4)**

At the beginning of your journey, you will define the destination: the objectives and use case of your project. In the early weeks, take time to ponder what problem you want to solve with image processing. Think about why you have chosen this particular topic and how you plan to tackle the challenges ahead. During this stage, you are expected to create a project proposal—a clear narrative of your objectives, the use case you have selected, and a step-by-step plan for reaching your goal. Do discussions with your Teaching Assistant on the effort in order to help you make a better estimation of your effort.

## **Marking Criteria:**

- **Mark 5:**
  - Provides a basic objective and a simple, high-level plan.
  - Limited or no evidence of TA input or preliminary research.
- **Marks 6–7:**
  - Clearly defined objectives with a structured plan that lists main tasks.
  - Some initial research and evidence of TA consultation.
- **Mark 8-9:**
  - A well-reasoned rationale that explains the project's purpose and approach in detail.

- A detailed plan that identifies tasks, potential challenges, and ways to address them.
- **Marks 10:**
  - A thorough rationale with clear explanations of all aspects of the project.
  - A comprehensive and organized plan that demonstrates careful thought and planning.

## Phase 2: First Phase Implementation (Weeks 5–10)

With your plan in place, you now set off to build the backbone of your project. During these weeks, you will translate your ideas into a tangible, working prototype. This phase is about creating a version of your project that demonstrates the core functionalities you planned. As you write your code, think of it as the structure of a building: it doesn't have to be perfect on the first try, but it should reflect the design you envisioned.

### Marking Criteria:

- **Mark 5:**
  - A basic working implementation covering essential functionalities.
  - Minimal explanation of design choices.
- **Marks 6–7:**
  - A nearly complete implementation with most core functions working.
  - A clear explanation of key design decisions.
- **Mark 8-9:**
  - A complete and well-organized implementation that follows the planned tasks.
  - Sufficient explanation of design choices, focusing on how each decision contributes to achieving project objectives.
- **Marks 10:**
  - A robust and optimized implementation that meets all the planned objectives.
  - Clear and logical explanation of design decisions without overreliance on extensive documentation.

## Phase 3: Test Phase (Weeks 11–12)

As you approach the middle of your journey, it's time to put your project to the test. In this phase, you will challenge your implementation by running it through a variety of test cases. Imagine yourself as an explorer examining every pathway: you want to ensure that your solution works well under different conditions, and you need to document both the successes and the obstacles you encounter.

During the test phase, you should develop a test plan that explains your approach and list the scenarios you are testing. Gather your observations, noting down any errors or performance issues, and compile these into a summary report.

## Marking Criteria:

- **Mark 5:**
  - Conducts basic testing with a few test cases and provides a short summary of results.
- **Marks 6–7:**
  - Performs detailed testing across multiple scenarios.
  - Provides a clear summary that identifies several issues.
- **Mark 8-9:**
  - Completes comprehensive testing that covers common scenarios and edge cases.
  - Provides a well-organized summary of issues along with initial suggestions for improvement.
- **Marks 10:**
  - Conducts extensive and systematic testing with clear performance metrics.
  - Offers a detailed analysis of issues and well-founded recommendations for improvements.

## Phase 4: Incremental Improvement (Weeks 13–14)

Having identified the areas where your project can be strengthened, you now refine and improve your implementation. Think of this stage as revisiting a rough draft and polishing it to perfection. Based on the test phase, you will update your code to address the issues you discovered. Your focus should be on enhancing performance in at least two specific scenarios, and you should explain how these improvements resolve the earlier issues. You will provide an updated version of your code along with a summary of the changes made.

### Mark 5:

- Implements minimal improvements with a basic explanation.

### Marks 6–7:

- Provides noticeable improvements in performance on key aspects.
- Offers a clear summary of the changes made and their impact.

### Mark 8-9:

- Demonstrates significant improvements that directly address the issues identified in testing.
- Provides a detailed summary with performance comparisons.

### Marks 10:

- Delivers substantial enhancements that lead to measurable improvements.

- Offers a comprehensive explanation of changes and a clear, objective performance analysis.

## Final Deliverables

At the end of your project, you will present the final product of your journey. Your GitHub repository should showcase the complete codebase, reflecting a clear progression through the project phases. Along with this, a final report will provide an overarching narrative of your journey—from the initial rationale to the final improvements. The report should briefly recount the decisions made in each phase, the challenges you encountered, and the lessons learned along the way.

## Final Mark

The final mark will be computed given the following formula:

$$0.2 * \text{Phase 1} + 0.5 * \text{Phase 2} + 0.2 * \text{Phase 3} + 0.1 * \text{Phase 4}$$