Islamic University of Technology

Visual Programming Lab

CSE 4402

Final Project Requirements

Author:
Ishmam Tashdeed
CSE, IUT

Contents

1 Introduction

Welcome to the lab final project documentation for CSE 4402: Visual Programming Lab. This document serves as your comprehensive guide for completing the culminating project of the course. It outlines all essential requirements, expectations, and guidelines to help you successfully navigate each stage of your final project. The final project is designed to allow you to apply the concepts and techniques you have learned throughout the semester to develop a complete, functional, and well-documented visual application. Through this project, you will demonstrate your understanding of core visual programming principles, event-driven logic, and user interface design. Please read through the entire document carefully and refer back to it regularly as you progress. Your ability to work effectively as a team, plan and design creatively, and demonstrate technical competence will be key factors in your evaluation.

2 Team Formation

In this course, you are expected to collaborate in small teams to complete the final project. Team-based work simulates real-world software development environments and helps cultivate essential skills such as communication, coordination, and division of labor.

Each team must consist of **no more than three students**. While teams of two or even one student may be allowed under exceptional and well-justified circumstances, teams exceeding three members are strictly prohibited, without exception. Only students who are officially registered for the CSE 4402: Visual Programming Lab course are permitted to participate in a team. Students from different lab sections are allowed to form a team, provided that all members are enrolled in the course during the current semester and no collisions with other labs occur.

Each team member is expected to take ownership of specific aspects of the project—whether related to design, implementation, documentation, or testing—and will be held individually accountable for their contributions. While collaboration and peer support are both encouraged and expected, evaluation will be based on individual effort in addition to team outcomes. This means that all team members must be actively engaged in the project lifecycle and should maintain transparency regarding their assigned responsibilities.

During evaluation, each student may be asked to explain the components they were responsible for and demonstrate a clear understanding of the project as a whole. Teams are therefore advised to distribute work fairly and ensure that all members are consistently involved throughout the development process. It is the responsibility of each team to ensure internal coordination and equitable workload distribution. Conflicts or difficulties within teams should be reported to the course instructor as early as possible to avoid disruption to project progress.

3 Project Scope

The final project is designed to assess your understanding and practical application of visual programming concepts using Java or Java-adjacent frameworks and libraries. Your task is to design and implement a complete application with a Graphical User Interface (GUI) that demonstrates your proficiency in key areas such as GUI components, event handling, object-oriented programming (OOP), and other relevant topics seen throughout the lab sessions.

3.1 Sustainable Development Goals

As part of IUT's commitment to academic excellence and social responsibility, and in accordance with the directives of the Academic Quality Assurance (AQA) Committee, all final projects for CSE 4402: Visual Programming Lab are required to demonstrate relevance to **at least one** of the United Nations Sustainable Development Goals (SDGs).

The Sustainable Development Goals (SDGs) are a collection of 17 global goals established by the United Nations in 2015 as part of the 2030 Agenda for Sustainable Development. These goals represent a universal call to action to end poverty, protect the planet, and ensure peace and prosperity for all. Each goal is accompanied by specific targets aimed at addressing the world's most pressing social, economic, and environmental challenges. These goals are:

- No poverty
- Zero Hunger

- Good Health and Well-being
- Quality Education
- Gender Equality
- Clean Water and Sanitation
- Affordable and Clean Energy
- Industry, Innovation and Infrastructure
- Reduced Inequalities
- Sustainable Cities and Communities
- Responsible Consumption and Production
- Climate Action
- Life Below Water
- Life On Land
- Peace, Justice and Strong Institutions
- Partnerships for the Goals

For detailed descriptions, you are encouraged to visit: https://sdgs.un.org/goals

Your selected project must align with **at least one** SDG in a meaningful way. This means the core purpose of your application should aim to support or promote one of the goals, either directly or indirectly. You do not need to solve a global issue—but your project should demonstrate awareness and relevance to an aspect of sustainable development. Here are a few examples of how your project might align:

- A health tracking or mental well-being app may align with Goal 3: Good Health and Well-being.
- An educational game or platform could support Goal 4: Quality Education.
- An app for tracking trash cans around the city supports Goal 11: Sustainable Cities and Communities

By integrating the SDGs into your final project, you are not only practicing responsible innovation but also aligning your technical skills with broader global challenges—a mindset that reflects the values of both IUT and the global engineering and computing community.

3.2 Choosing The Right Project

The scope of the project does not need to be vast or overly complex. Rather, you are encouraged to select a real-life problem—something that addresses a practical, day-to-day challenge. Whether simple or sophisticated in appearance, the emphasis should be on creating solutions that are meaningful, relevant, and technically sound. **Projects that merely replicate common, uninspired ideas will not be entertained**.

In particular, we are **banning "management system"**—style projects unless they demonstrate a genuinely novel angle or are tailored to solve a specific and compelling real-world problem. Creativity, originality, and thoughtfulness are highly valued.

3.3 On Scope Realism

One of the most common mistakes students make is being overly ambitious at the proposal stage—attempting to build applications with features far beyond the time and resources available. This often leads to a need for repeated scope reduction, resulting in a final product that bears little resemblance to the initial concept. On the other hand, projects with an overly narrow or trivial scope (e.g., simple calculators or single-feature tools) risk being evaluated as underwhelming and too simple. You are advised to be realistic and balanced in your planning. This lab project is just one component of a full-semester workload that includes multiple other courses. As a rule of thumb, each team member is expected to invest 10–12 hours per week into the project. Plan your project accordingly.

3.4 Original Work

To ensure that the project reflects your own understanding and effort, we strongly discourage an excessive reliance on third-party APIs. While you may use libraries and tools for development (e.g., JavaFX, Scene Builder, Java Swing, Spring Boot), the core implementation must be authored by

your team. Reusing pre-existing codebases or integrating pre-built solutions undermines the educational purpose of this project and may negatively impact your evaluation.

4 Some Example Projects

As previously emphasized, strict adherence to the defined project scope is imperative for the success and manageability of your final project. While innovation and creativity are encouraged, the project must remain technically feasible, conceptually sound, and realistically achievable within the semester's constraints. To help you conceptualize and plan your own project, a list of example projects is provided below. These examples are selected to reflect the expected scope, complexity, and real-world relevance we are looking for. It is essential to note that you are **NOT EXPECTED TO MATCH OR EXCEED** the capabilities of these projects. They are included for inspiration and guidance only, and should serve as a reference point to help you shape your unique and meaningful idea. Some of the following ideas are from the internet whereas some of them are exceptional projects from students who took this course last year:

- Automatic resume generator application that takes in your information and generates a resume from a chosen template. Similar to this application. This could support the goal of Reduced Inequalities.
- A program to change/update the internet ID being used in the IUT dormitory rooms. Similar to this application. This could support the goal of Responsible Consumption and Production.
- A intuitive Quran app to read the Quran. Similar to this application. This could support the goal of Quality Education.
- A fun visualizer app to learn solving a Rubik's cube. Similar to this application. This could support the goal of Quality Education.
- An interactive visualizer app to learn how different CS algorithms work. Similar to this application. This could support the goal of Quality Education.

Pay attention to how these projects maintain their GitHub page. We **DO NOT** expect you to match/replicate these projects. But your project idea should aim to solve a problem similar to how these exemplary projects aim to do.

5 Project Proposal Presentation

The first critical milestone of your final project journey is the development and presentation of a comprehensive project proposal. This phase sets the foundation for your entire project and must be approached with thorough planning and clear articulation of your ideas. Each team is required to present their proposal in a concise and structured manner. The exact date of the proposal presentation will be announced in due time. Teams should be fully prepared to deliver their proposal presentation as per the guidelines outlined in the following parts.

5.1 Presentation Format

- **Duration**: Each team will be allotted six minutes for the presentation. An additional 30 seconds may be granted if necessary to conclude.
- **Participation**: It is mandatory for all team members to actively participate in the presentation. This ensures equal contribution and familiarity with all aspects of the proposed work.
- **Visual Aid**: Teams are required to use presentation slides to effectively communicate their ideas. The presentation should be clear, focused, and professionally structured.

5.2 Requirements for Project Proposal

- **Motivation**: Clearly explain the problem you aim to solve and why it is worth addressing. Provide background, relevance, and the real-world context that justifies the selection of your project idea. Your motivation should align with the themes of practical impact, creativity, or social value, and, where possible, connect to a Sustainable Development Goal (SDG).

- **Goals and Objectives**: Define the intended outcomes of your project. What do you aim to accomplish by the end of the semester? Break down your broader goals into specific, measurable, and achievable objectives. These will later serve as benchmarks for evaluating the success and completeness of your final submission.
- **Differentiation from Existing Solutions**: Identify any existing systems or applications that tackle a similar problem. Clarify how your approach is distinct or improved, whether in terms of user experience, design, functionality, scalability, or accessibility. This section is key to demonstrating originality and technical insight.
- **Features**: List the proposed features of your application. Your features must be functional, intuitive and helpful for the end user. You can converse with the stakeholders of your project to find suitable features.
- **Technical Tools**: List the programming languages, frameworks, libraries, and tools you plan to use (e.g., JavaFX, Scene Builder, databases, APIs if approved). Justify your choices based on the project requirements and your team's familiarity with these tools.
- **Timeline**: Create a tentative Gantt Chart or timeline that maps out major project milestones from proposal approval to final submission. Include deadlines for stages such as design, implementation, testing, UI refinement, and documentation. A well-structured timeline will reflect your team's project planning and task distribution capabilities.

Treat it as a professional pitch — be prepared, practice ahead, and make sure your presentation reflects the collective understanding and enthusiasm of the entire team. Failure to present a compelling, realistic, and well-planned proposal may result in the rejection of the project idea, requiring resubmission within a limited timeframe.

5.3 Project Proposal Evaluation

Your proposal will be evaluated according to the following key dimensions:

- **Relevance and Motivation (20%)**: How clearly does the team define the problem? Is the motivation compelling, well-articulated, and grounded in real-world context? Does the idea show purpose or social value (e.g., through alignment with SDGs)?

- **Goals and Feasibility (10%)**: Are the objectives specific, realistic, and achievable within the time and resource constraints? Is the project appropriately scoped for the semester?
- **Originality and Differentiation (10%)**: Does the proposed solution offer something new or improved compared to existing alternatives? Is there clear evidence of creativity, innovation, or unique problem-solving?
- **Features (20%)**: Are the features relevant and sufficient? Does the feature set reflect the opinions of the stakeholders?
- **Technical Preparedness (10%)**: Has the team selected appropriate tools, frameworks, and technologies? Is there a reasonable match between the chosen tools and the nature of the project?
- **Timeline and Planning (10%)**: Is the proposed timeline structured, complete, and practical? Are tasks and milestones clearly defined to ensure consistent progress throughout the semester?
- **Presentation Quality and Team Involvement (20%)**: Was the presentation clear, well-organized, and professional? Did all team members contribute meaningfully to the presentation, demonstrating shared understanding and responsibility?

6 Final Project Presentation

The final project presentation marks the culminating milestone of your work in CSE 4402: Visual Programming Lab. The final project will contribute to **50%** of your total grade in this course, the remaining amount is from continuous evaluation of the lab tasks. For the final project, **40%** of the grade depends on the project proposal while the remaining depends on the final presentation. Teams should be fully prepared to deliver their final presentation as per the guidelines outlined in the following parts.

6.1 Presentation Format

- **Duration**: Each team will be allotted six minutes for the presentation. An additional 30 seconds may be granted if necessary to conclude.

- **Participation**: It is mandatory for all team members to actively participate in the presentation. This ensures equal contribution and familiarity with all aspects of the proposed work.

- **Visual Aid**: Teams are required to use presentation slides to effectively communicate their ideas. You are expected to give a **complete demonstration** of your application as well within the allocated time. The presentation should be clear, focused, and professionally structured.

6.2 Requirements for Final Project Presentation

- **Project Overview**: Briefly summarize the objective and purpose of your application. Reiterate the problem you set out to solve and provide a quick recap of your motivation.

- **Features and Functionality**: Highlight the key features of your application. Focus on what you have actually implemented, not just what was planned.

- **Visual Design and User Experience**: Explain your GUI design decisions. How did you ensure usability, responsiveness, or clarity in the interface? Showcase how visual programming concepts were applied thoughtfully.

- **Project Goals and Achievement Evaluation**: Reflect on the original goals and objectives outlined in your proposal. Clearly state which goals were fully achieved, which were partially met, and which were adjusted. Justify any deviations or compromises with proper reasoning.

- **Version Control**: You must use Git for version control while building your project and use GitHub to store and manage the code, as well as track and control changes. You must follow proper software development etiquette while using Git/GitHub utilizing commit comments, pull requests etc.

- **Documentation**: You must provide detailed documentation for your code. Include comments, explanations, and any external resources/references used during the development process. You may use *Javadoc* to document your project if you feel like it. You will have to submit

a full report documenting your project. The details for that will be circulated soon.

- **Demonstration**: Finally, a complete demonstration of your application has to be performed showing off the different features.

6.3 Final Presentation Evaluation

Your final presentation will be evaluated according to the following key dimensions:

- **Functionality and Completeness (25%)**: Does the application run successfully and perform the intended functions? Are the key features as described in the proposal fully implemented? Does the application operate reliably during the live demonstration?
- **User Interface and Usability (15%)**: Is the interface intuitive and user-friendly? Does the design demonstrate clarity, responsiveness, and consistency in user experience?
- **Code Quality (10%)**: Is the code well-structured, readable, and appropriately documented? Are events captured and handled effectively, enhancing user interaction using visual programming concepts? Is proper version controlling being used?
- **Presentation and Communication (25%)**: Was the presentation well-organized, clear, and engaging? Did the team effectively communicate the purpose, features, and technical details of the project within the allotted time?
- **Goals and Objectives (15%)**: Has the team selected clearly completed the original goals and objectives outlined in the initial proposal?
- **Documentation (10%)**: Has the team submitted the report documenting different aspects of the development process?

7 Conclusion

This documentation serves as your guide throughout the final project development process. Please adhere to the outlined requirements and guidelines

to ensure a successful completion of the course. If you have any questions or require clarification, do not hesitate to seek assistance by emailing me at: ishmamtashdeed@iut-dhaka.edu. Best of luck with your project!