Agile Project Description Document

Project Title

Agile Development of a Graphical Calculator Using Pygame

Course Context

Course: The Agile Professional (SE 3122)
Frameworks Used: Scrum + Kanban
Duration: 2 Weeks (14 Days)
Team Size: 9 Members
Tools: Python, Pygame, GitHub, GitHub Projects

---

1. Project Overview

This project involves the **design and development of a graphical calculator application using Python and the Pygame library**, executed using **Agile methodologies**. The project is simulated in a real-world Agile environment by applying **Scrum for sprint planning and reviews**, and **Kanban for task tracking and workflow visualization** using **GitHub Projects**.

The calculator will support basic arithmetic operations through a graphical user interface (GUI), reinforcing both **technical programming skills** and **Agile team collaboration practices**.

---

2. Project Objectives

2.1 Technical Objectives

- Develop a fully functional calculator using **Pygame**
- Implement basic arithmetic operations (+, −, ×, ÷)
- Design an intuitive graphical user interface (buttons, display screen)
- Handle user input via mouse and keyboard events
- Ensure error handling (e.g., division by zero)

2.2 Agile Learning Objectives

- Apply **Scrum ceremonies** (Sprint Planning, Daily Stand-ups, Review, Retrospective)
- Use **Kanban boards** to visualize workflow
- Collaborate using **GitHub and GitHub Projects**
- Practice backlog creation, sprint execution, and incremental delivery

---

3. Agile Framework Selection and Justification

3.1 Scrum

Scrum is used to structure the project into **time-boxed sprints**, ensuring incremental delivery and continuous feedback. It is suitable because: - The project has a clear goal and short timeline - Work can be broken into small, deliverable increments - Frequent inspection and adaptation are required

3.2 Kanban

Kanban complements Scrum by: - Visualizing task progress - Limiting work-in-progress (WIP) - Improving team transparency

GitHub Projects serves as the **Kanban board** with columns such as *Backlog*, *To Do*, *In Progress*, *Review*, and *Done*.

---

4. Team Structure and Roles (9 Members)

Role	Responsibilities
Product Owner (1)	Defines features, manages product backlog, prioritization
Scrum Master (1)	Facilitates Scrum events, removes blockers
Developers (5)	Coding logic, GUI, event handling
UI/UX Designer (1)	Button layout, screen design
Tester & QA (1)	Testing, bug reporting, validation

---

5. Tools and Technologies

- **Programming Language:** Python
- **Library:** Pygame
- **Version Control:** GitHub
- **Project Management:** GitHub Projects (Kanban)
- **Communication:** WhatsApp / Discord / Google Meet

---

6. Product Backlog (High-Level)

6.1 User Stories

ID	User Story	Priority
US1	As a user, I want a calculator window to open	High
US2	As a user, I want clickable number buttons	High
US3	As a user, I want arithmetic operation buttons	High
US4	As a user, I want to see inputs on a display screen	High
US5	As a user, I want correct calculation results	High
US6	As a user, I want a clear/reset button	Medium
US7	As a user, I want keyboard input support	Medium
US8	As a user, I want error messages for invalid operations	Medium
US9	As a user, I want a visually appealing interface	Low
US10	As a user, I want the app to exit cleanly	Low

---

7. Sprint Planning (2-Week Timeline)

Sprint Structure

- **Sprint 1:** Core functionality & basic UI (Week 1)
- **Sprint 2:** Enhancements, testing & final delivery (Week 2)

Each sprint lasts **1 week**.

---

8. Sprint 1 – Core Calculator Development (Week 1)

Sprint Goal

Deliver a working calculator with basic arithmetic operations and a simple GUI.

Sprint Backlog – Sprint 1

Task	Assigned Role
Setup GitHub repository	Scrum Master
Create project Kanban board	Product Owner

Task	Assigned Role
Initialize Pygame window	Developer
Design calculator layout	UI/UX Designer
Implement number buttons	Developer
Implement operators (+ − × ÷)	Developer
Display input/output	Developer
Daily stand-ups	All

Deliverables

- Running Pygame window
- Functional buttons
- Correct arithmetic calculations

---

9. Sprint 2 – Enhancements & Testing (Week 2)

Sprint Goal

Improve usability, stability, and finalize the application.

Sprint Backlog – Sprint 2

Task	Assigned Role
Implement clear/reset function	Developer
Keyboard input support	Developer
Error handling (division by zero)	Developer
UI polishing (colors, fonts)	UI/UX Designer
Testing & bug fixing	QA Tester
Code refactoring	Developers
Documentation (README)	Product Owner
Sprint review & retrospective	All

Deliverables

- Fully tested calculator
- Improved UI
- GitHub documentation

10. Kanban Workflow (GitHub Projects)

Columns Used: - Backlog - To Do - In Progress - Review - Done

Tasks move across columns daily, ensuring transparency and continuous delivery.

---

11. Scrum Ceremonies Applied

- **Sprint Planning:** Backlog selection and sprint goal definition
- **Daily Stand-ups:** Short daily sync meetings
- **Sprint Review:** Demonstration of working calculator
- **Sprint Retrospective:** Reflection on what went well and improvements

---

12. Challenges and Risk Management

Challenge	Mitigation
Time constraints	Short sprints & clear priorities
Merge conflicts	Feature branches & pull requests
Skill differences	Pair programming
Bugs	Continuous testing

---

13. Final Deliverables

- Pygame Calculator Application
- GitHub Repository with commits
- GitHub Project Board (Kanban)
- Final Project Report
- PowerPoint Presentation

---

14. Conclusion

This project demonstrates the effective use of **Agile methodologies** in a small-scale software development environment. By combining **Scrum for structure** and **Kanban for visualization**, the team achieves rapid development, collaboration, and continuous improvement while delivering a functional calculator application using Pygame.