**1. Project Overview**

**Goal:** Build a simple yet functional calculator application in Java that supports basic arithmetic operations (addition, subtraction, multiplication, and division) with a graphical user interface (GUI).

**2. Timeline**

**Estimated project duration:** 2 weeks

**3. Milestones & Tasks**

**Milestone 1: Project Setup & Initial Planning**

* **Task 1.1:** Define project scope & requirements
  + *Time:* 0.5 day
  + *Deliverable:* Project Requirements Document
* **Task 1.2:** Set up project repository and development environment
  + *Time:* 1 day
  + *Deliverable:* Initialized Git repository with README file
* **Task 1.3:** Design the basic UI layout and wireframes
  + *Time:* 1 day
  + *Deliverable:* UI Wireframes and Design Draft

**Milestone 2: Backend Development**

* **Task 2.1:** Implement core calculator logic (addition, subtraction, multiplication, division)
  + *Time:* 2 days
  + *Deliverable:* Working Java class with arithmetic methods
* **Task 2.2:** Implement error handling (divide by zero, invalid inputs)
  + *Time:* 1 day
  + *Deliverable:* Exception handling functionality

**Milestone 3: Frontend Development**

* **Task 3.1:** Develop the GUI using Java Swing/JavaFX
  + *Time:* 2 days
  + *Deliverable:* Basic UI layout with buttons and display
* **Task 3.2:** Connect UI components to backend calculator logic
  + *Time:* 2 days
  + *Deliverable:* Functional calculator with working buttons
* **Task 3.3:** Implement additional features (clear button, decimal support)
  + *Time:* 1 day
  + *Deliverable:* Enhanced functionality

**Milestone 4: Testing & Bug Fixing**

* **Task 4.1:** Write unit tests for arithmetic operations
  + *Time:* 1 day
  + *Deliverable:* JUnit test cases covering core functions
* **Task 4.2:** Perform GUI testing
  + *Time:* 1 day
  + *Deliverable:* Functional GUI with responsive buttons
* **Task 4.3:** Fix identified bugs and refine UI
  + *Time:* 1 day
  + *Deliverable:* Improved UI and bug-free functionality

**Milestone 5: Deployment & Final Review**

* **Task 5.1:** Package and document the application
  + *Time:* 1 day
  + *Deliverable:* Packaged JAR file and setup guide
* **Task 5.2:** Collect feedback and make final adjustments
  + *Time:* 0.5 day
  + *Deliverable:* Finalized application

**4. Deliverables**

1. **Project Requirements Document** – Detailed description of functionality
2. **Source Code** – Full Java source code (GUI and logic)
3. **UI Wireframes/Design** – Sketch or digital mockup of UI
4. **Executable Application** – Packaged JAR file for distribution
5. **Test Suite** – JUnit tests for core functionality
6. **Final Project Report** – Summary of project and challenges faced

**5. Risks and Mitigation Plans**

* **Risk:** UI not working as expected
  + *Mitigation:* Incremental testing and feedback loops
* **Risk:** Arithmetic logic errors
  + *Mitigation:* Unit testing and validation of outputs
* **Risk:** Compatibility issues with Java versions
  + *Mitigation:* Testing on different Java versions before release

**6. Project Schedule Overview**

|  |  |  |
| --- | --- | --- |
| **Week** | **Task** | **Duration** |
| 1 | Project setup & backend logic | 4 days |
| 1-2 | Frontend development | 4 days |
| 2 | Testing & bug fixing | 3 days |
| 2 | Deployment & documentation | 3 days |

This structured plan ensures a smooth development process and timely completion of the project.