**Developing a Scientific Calculator with Computation History**

**Members:**

**Agonoy, Alexis Greko F.**

([alexisagonoy0107@gmail.com](mailto:alexisagonoy0107@gmail.com))

**Bande, Meliza H.**

([melizahonorio0@gmail.com](mailto:melizahonorio0@gmail.com))

**Ragiles, Tristan Bren V.**

([tristan082300@gmail.com](mailto:tristan082300@gmail.com))

**Torralba, Laiza Marie P.**

**(**[**laizatorralba2@gmail.com**](mailto:laizatorralba2@gmail.com)**)**

**Valdez, Kate Ashley R.**

([kateashleyvaldez19@gmail.com](mailto:kateashleyvaldez19@gmail.com))

March 2024

1. **Introduction**

In today's fast-paced digital landscape, the demand for efficient tools is evident, particularly in the realm of mathematics and scientific computations. Traditional scientific calculators, while widely used, lack a crucial feature: the ability to maintain a computation history. This absence poses challenges for professionals and students alike, leading to loss of context in complex calculations, difficulty in error detection and debugging, and challenges in auditing and documentation.

Customer feedback and market trends emphasize the importance of this feature, highlighting a clear demand for scientific calculators with computation history functionality. Addressing these needs by developing a Scientific Calculator with Computation History is not only a wise investment but also a necessary step in meeting the evolving demands of users across various domains.

1. **Current Process Analysis**
2. **Proposed Improvement**
3. **Benefits & Impact**
4. **Project Plan & Timeline**

**Phase 1: Planning and Requirements Gathering**

Duration: 2 days

**Milestones:**

1. Define project scope and objectives.
2. Conduct market research to identify existing scientific calculators and features.

**Resources Needed:**

1. Project Manager
2. Programming Languages: HTML, CSS, Javascript.
3. Integrated Development Environments (IDEs): Visual Studio.
4. Version Control: Git

**Phase 2: Design and Prototyping**

Duration: 1 week

**Milestones:**

1. Design the user interface (UI) for the calculator, including buttons, layout, and display.
2. Create wireframes or mockups to visualize the interface.
3. Design the architecture for computation history storage and retrieval.

**Resources Needed:**

1. UI/UX designer (HTML & CSS)
2. Prototyping tools (Figma)
3. User testing participants

**Phase 3: Development**

Duration: 2 weeks

**Milestones:**

1. Develop core calculator functionality (basic arithmetic operations).
2. Implement scientific functions (trigonometric, logarithmic, etc.).
3. Integrate computation history feature.
4. Perform thorough testing and debugging.

**Resources Needed:**

1. Software developers (front-end and back-end)
2. Database engineer
3. Testing team

**Phase 4: Testing and Quality Assurance**

Duration: 1-2 weeks

**Milestones:**

1. Conduct functional testing of all calculator features.
2. Perform compatibility testing on different platforms and devices.
3. Address any bugs or issues identified during testing.

**Resources Needed:**

1. Testing tools
2. Devices for compatibility testing

**Phase 5: Deployment and Launch**

Duration: 1 week

**Milestones:**

1. Deploy the scientific calculator to a web server.
2. Perform final checks and optimizations.
3. Launch the project to the target audience.

**Resources Needed:**

1. Deployment tools
2. **Evaluation & Risk Assessment**
3. **Conclusion**

Our proposal aims to develop a Scientific Calculator with Computation History, filling a crucial void in traditional calculators by introducing a feature that retains the history of calculations. This enhancement addresses the inefficiencies and challenges associated with the absence of computation history functionality, such as reduced efficiency, error detection difficulties, and auditing complexities. By leveraging approaches like utilizing a database engine to store recent calculations, our solution promises significant benefits across education, business, and research sectors.

It will elevate user experience, improve learning outcomes, enhance accuracy and efficiency, and foster collaboration among users. With a detailed project plan encompassing all phases from planning to deployment, alongside a thorough evaluation and risk assessment strategy, our proposal offers a robust framework for the development of this transformative tool. In essence, the Scientific Calculator with Computation History represents a paradigm shift in mathematical computations, aligning with evolving user demands and driving advancements in technology to meet the needs of professionals, students, and enthusiasts alike.