CIS641 Calculator Project

Vivi Hoang and Pratik Shrestha
Dec. 16, 2020

Team members

Vivi

Pratik

Overview

- Project description
- Time goals / Progress since midterm
  - Remaining issues
- Change management plan
- Testing
- Deployment
- Demo of final implementation

Project description

- Calculator app
  - Arithmetic
  - Trigonometry
  - Unit conversion
- Initial proposal rationale
  - Saving time
  - Improving accuracy

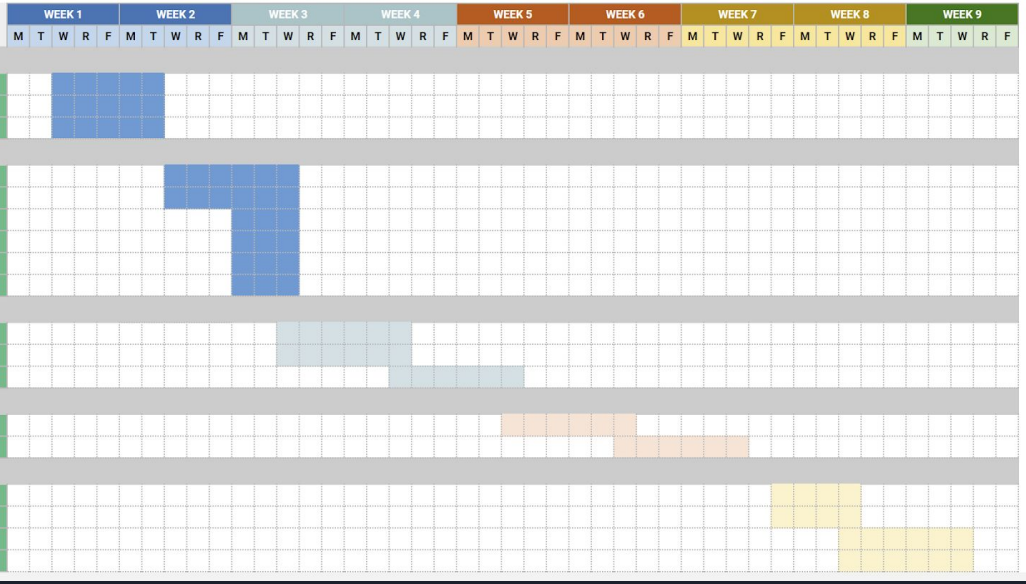
Timeline / Progress

https://docs.google.com/spreadsheets/d/1fueWszEquuGhWwA7LTNIGlknG-qABH2ABIWLH1ITMMg/edit?usp=sharing

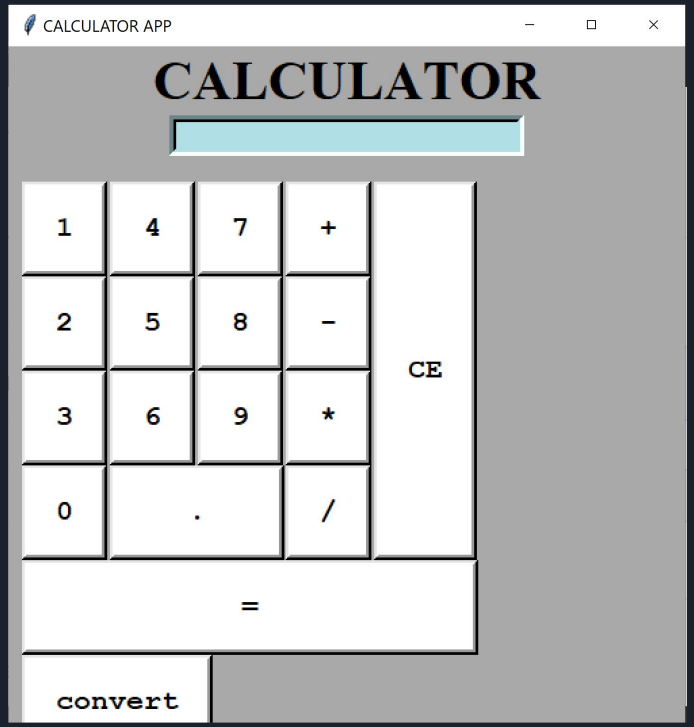
Calculator Timeline

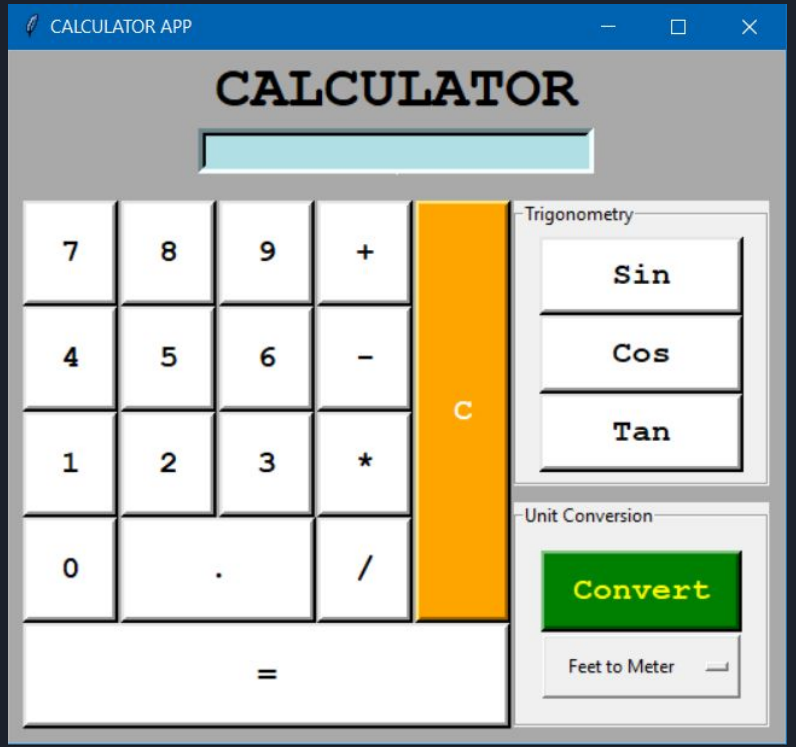
PROJECT TITLE	Calculator	DATE	10/28/2020
PROJECT TEAM MEMBERS	Vivi Hoang and Pratik Shrestha		

TASK NUMBER	TASK TITLE	TASK OWNER	START DATE	DUE DATE	DURATION	PCT OF TASK COMPLETE
1	**Phase 1: Initial program**					
1.1	Set up Python and Anaconda coding environment	Vivi and Pratik	10/14/20	10/20/20	7	100%
1.2	Create basic Python program	Vivi and Pratik	10/14/20	10/20/20	7	100%
1.3	Research tkinter	Vivi and Pratik	10/14/20	10/20/20	7	100%
2	**Phase 2: Midterm Version 1**					
2.1	GUI implementation	Pratik	10/21/20	10/28/20	8	100%
2.2	Gantt chart	Vivi	10/21/20	10/28/20	4	100%
2.3	Review activity diagram	Vivi and Pratik	10/26/20	10/28/20	3	100%
2.4	Review use case description	Vivi and Pratik	10/26/20	10/28/20	3	100%
2.5	Midterm slides	Vivi and Pratik	10/26/20	10/28/20	3	100%
2.6	Testing	Vivi and Pratik	10/26/20	10/28/20	3	100%
3	**Phase 3: Version 2**					
3.1	Add unit conversion functionality	Pratik	10/29/20	11/4/20	8	100%
3.2	Implement class-based approach	Pratik and Vivi	10/29/20	11/4/20	8	100%
3.3	Test for bugs and edge cases	Vivi	11/4/20	11/11/20	8	100%
4	**Phase 4: Version 3**					
4.1	Add trigonometric functionality	Pratik and Vivi	11/11/20	11/18/20	8	100%
4.2	Test for bugs and edge cases	Vivi	11/18/20	11/25/20	8	100%
5	**Phase 5: Final**					
5.1	Finalize program	Pratik and Vivi	11/27/20	12/2/20	6	100%
5.2	Testing	Vivi	11/27/20	12/2/20	6	100%
5.4	Finalize documentation	Pratik and Vivi	12/2/20	12/9/20	8	100%
5.5	Finalize slides	Pratik and Vivi	12/2/20	12/9/20	8	100%



Progress at midterm vs final





Remaining issues of note

- After a calculation, the screen does not clear; it holds the old input.
- After first calculation, a Clear or “Invalid input” message, a space before cursor.
- Display allows for input beyond 25 characters; you just can’t see it.

Change management plan

- Rationale:
  - Improved accuracy and efficiency
  - Convenience
  - Cost
- Integration
- Training
- Issue Resolution

Testing

Test cases for Arithmetic Function

Inputs	Expected Output	Actual Output	Test Status
{<num><operator><num>}	valid input	Valid input	Passed
{<num><operator><num><operator>}	Invalid input	Invalid input	Passed
{<operator><num>} where <operator = '+', '-'>	valid input	Valid input	Passed
{<operator><num>} where <operator = '*','/'>	Invalid input	Invalid input	Passed
{<num><decimal><num><operator><num><decimal><num>}	valid input	Valid input	Passed

Test cases for Trigonometric Function

Inputs	Expected Output	Actual Output	Test Status
{<num><Func>}	Valid input	Valid input	Passed
{<operator><num>} where <operator = '+','-'>	Valid input	Valid input	Passed
{<operator><num>} where <operator = '*','/'>	Invalid input	Invalid input	Passed
{<num><decimal><num><Func>	Valid input	Valid input	Passed
{Tan(90) and Tan(-90)}	Invalid input	Invalid input	Passed

Test cases for Unit Conversion Function

Inputs	Expected Output	Actual Output	Test Status
{<num><FeetToMeter> OR <num><MeterToFeet>}	Valid input	Valid input	Passed
{<operator><num><FeetToMeter> OR <operator><num><MeterToFeet>}	Invalid input	Invalid input	Passed
{<num><decimal><num><FeetToMeter> OR {<num><decimal><num><FeetToMeter> }	Valid input	Valid input	Passed

Testing (Contd...)

Test cases for various function sequence

Operation sequence	Expected Output	Actual Output	Test Status
{<Arithmetic><Arithmetic>}	Valid Output	Valid Output	Passed
{<Arithmetic><CLEAR><Arithmetic>}	Valid Output	Valid Output	Passed
{<Arithmetic><UnitConversion>}	Valid Output	Valid Output	Passed
{<UnitConversion><UnitConversion>}	Valid Output	Invalid result	Failed
{<UnitConversion><CLEAR><UnitConversion>}	Valid Output	Valid Output	Passed
{<Arithmetic><Trigonometric>}	Valid Output	Valid Output	Passed
{<Arithmetic><CLEAR><Trigonometric>}	Valid Output	Valid Output	Passed
{<Trigonometric><Trigonometric>}	Valid Output	Invalid result	Failed
{<Trigonometric><CLEAR><Trigonometric>}	Valid Output	Valid Output	Passed

Deployment

Prerequisites:

- Python 3.7 or higher

Running from command line:

- Locate the file directory in which the project is stored.
- Type "python Calculator.py" in the command line to execute the application.

Running from Visual Studio Code

- The Python environment needs to be installed in VS Code.
- Open the project directory.
- Open the file "Calculator.py" and click Run and Debug (F5) in the left pane or click Run Code (Ctrl+Alt+N).

Can also be executed from other environments like Jupyter, Spyder , PyCharm etc.

Demonstration

Thank you !!

Any questions?