BILD 62 Final Project Rubric

Item	Needs Improvement	Meets some expectations	Meets all expectations
PROJECT INFORMATION (30)			
Project description *(10 points)*	Project description is very weak/absent	Project description only vaguely motivates the project or is too short	There is a project description in a Jupyter notebook that clearly & concisely explains the project and its motivation in 200-400 words
Team Contributions *(10 points)*	Team member contributions are not defined	Team member contributions are not well-defined or seem off balance	Each team member's contribution is balanced and well-defined
Reflection *(10 points)*	No personal reflection	Personal reflection is included but not thoughtful	Personal reflection is thoughtful, commenting on coding knowledge at the start of the class, and how the project challenged and expanded this knowledge
CODING CONTENT (90)			
Overall Approach *(20 points)*	Code / approaches / algorithms used for the project are inappropriate for the task, or are not original in any way	Code / approaches / algorithms used for the project somewhat make sense but could be optimized	The chosen code design & organization makes sense given the project topic outlined; code efficiently meets the challenge in the project
Functionality *(15 points)*	Multiple aspects of code do not work	Code is somewhat functional (small errors, inability to deal with particular use cases)	The code is functional - it does what it is supposed to, with no major errors or bugs, and works as intended
Contains three functions and/or methods *(15 points)*	Contains less than three working functions or methods	Contains three working functions or methods but these are not well-defined in scope or simply re-implement a built-in function in a commonly used Python package (e.g. np.mean())	Contains three working functions that accomplish clear, specific tasks. These functions can be inspired by code used in class but should not directly copy it.
Variables, Code Constructs, and Imports *(15 points)*	Multiple elements of code are inefficient, variables are poorly named, many	Code constructs are unnecessarily complex, long, inefficient, some packages imported that	Uses variables, control flow constructs and imports as needed and efficiently. Project demonstrates multiple coding elements from the

	packages imported that are never used	are never used. Project demonstrates few coding elements from class.	class.
Code Organization *(10 points)*	Code is not clearly organized.	Code is somewhat organized but could be improved.	Code is organized into modules & scripts as it makes sense. Front-facing Jupyter notebook is clean. Long functions are imported as scripts rather than in the Jupyter notebook.
Error Handling *(15 points)*	Very little error handling strategies within code	Code implements some error handling strategies but misses several cases	If data: code is resilient to multiple file types OR is clear about what data structure should be, giving clear messages if it does not meet specifications. If user interaction: code should handle various inputs and should be resilient to user error. Regardless: code should use some error catching strategies, such as assert, try/except, and unit tests
DOCUMENTATION & STYLE (20)			
Docstrings *(5 points)*	Functions/classes do not contain docstrings	Functions and classes contain docstrings that partially describe the function/class	There are docstrings on functions & classes that describe 1) code's function 2) parameters/inputs, 3) returns
Documentation *(5 points)*	Very few block or inline comments throughout project	There are some block or inline comments that helpfully explain the code	Any line of code that is not immediately self-explanatory to someone with a BILD62 level of Python knowledge is explained via inline or block comments and/or Markdown text
Code Style *(10 points)*	Spacing is inconsistent throughout & multiple style guide rules are not followed	Spacing is somewhat inconsistent and sometimes not following conventions, names are vague	There is good code layout - spacing between code segments, etc; names used are descriptive and follow naming conventions
PRESENTATION			
Final Showcase *(10 points)*	Student is not present at the final showcase (without emailing professor)	Student is present at the final showcase	Student is present at the final showcase and engaged in presenting their project as well as interacting with other projects