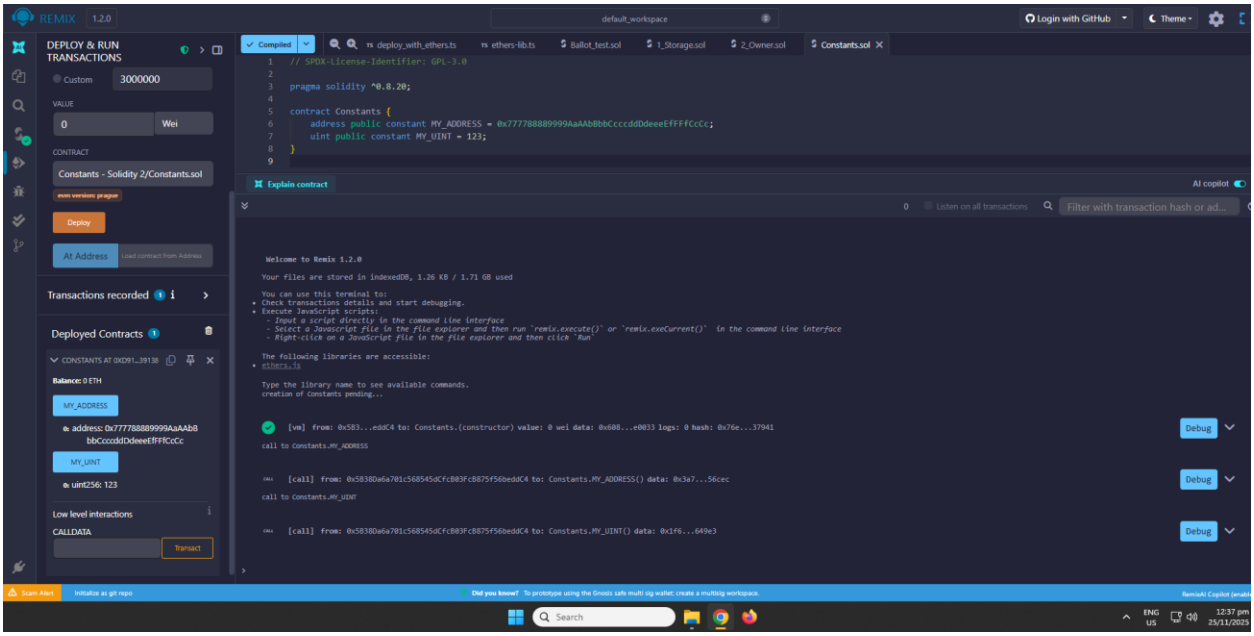
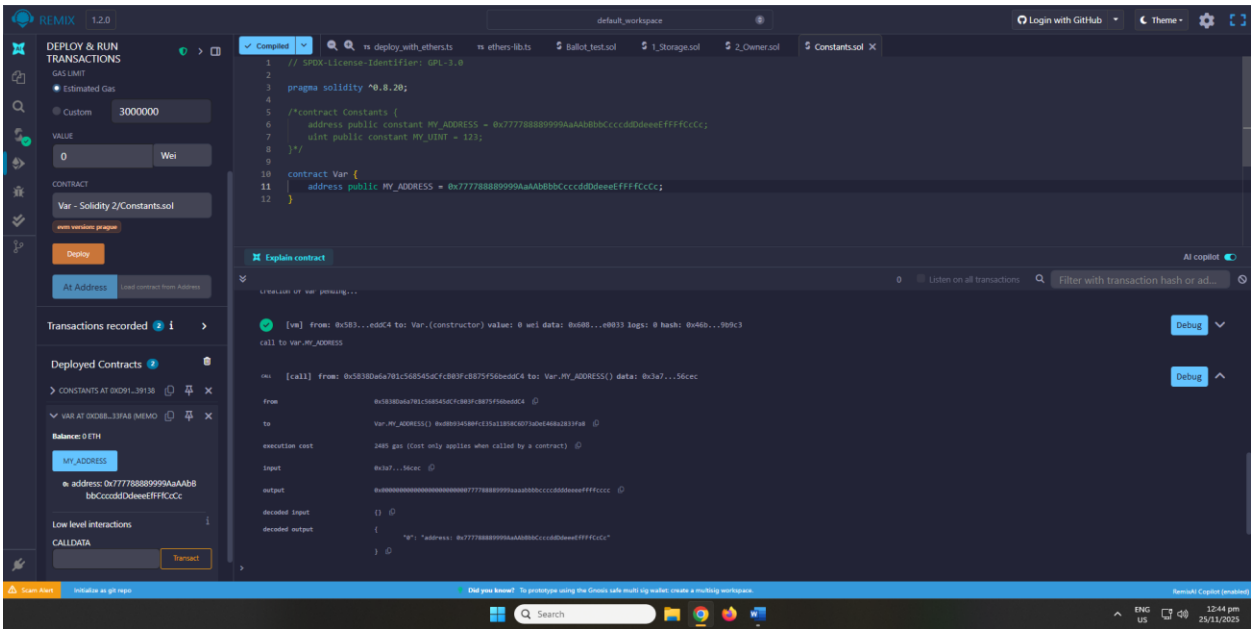
John Miguel P. Miclat

NW-301

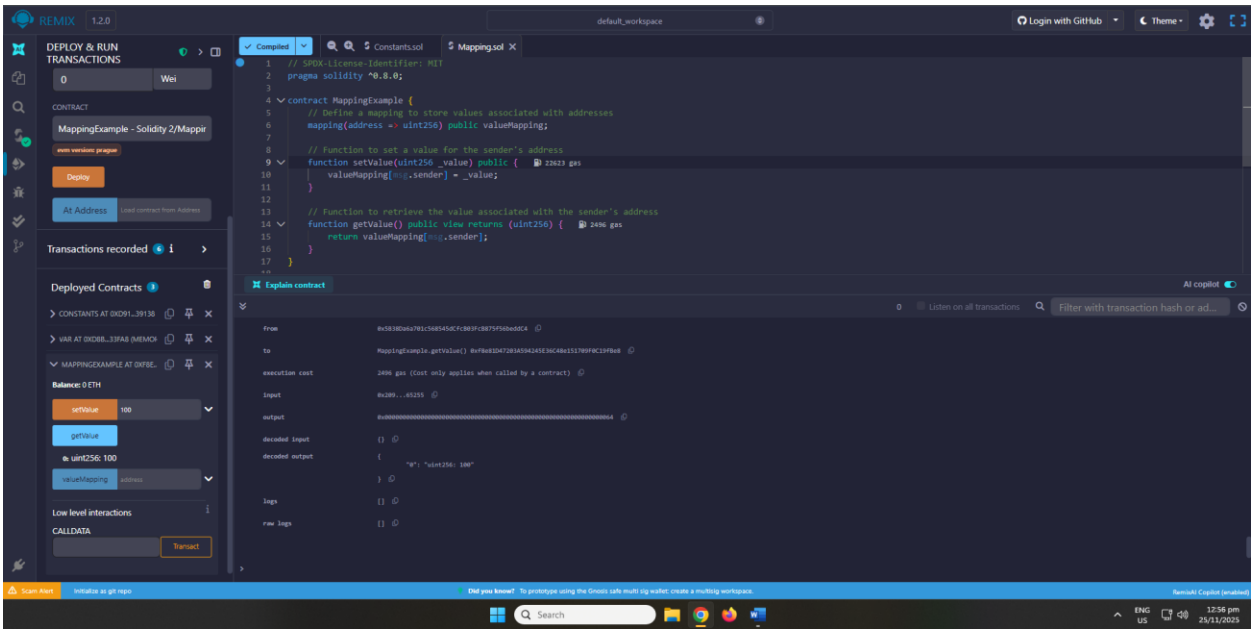
Week 3

Constants:

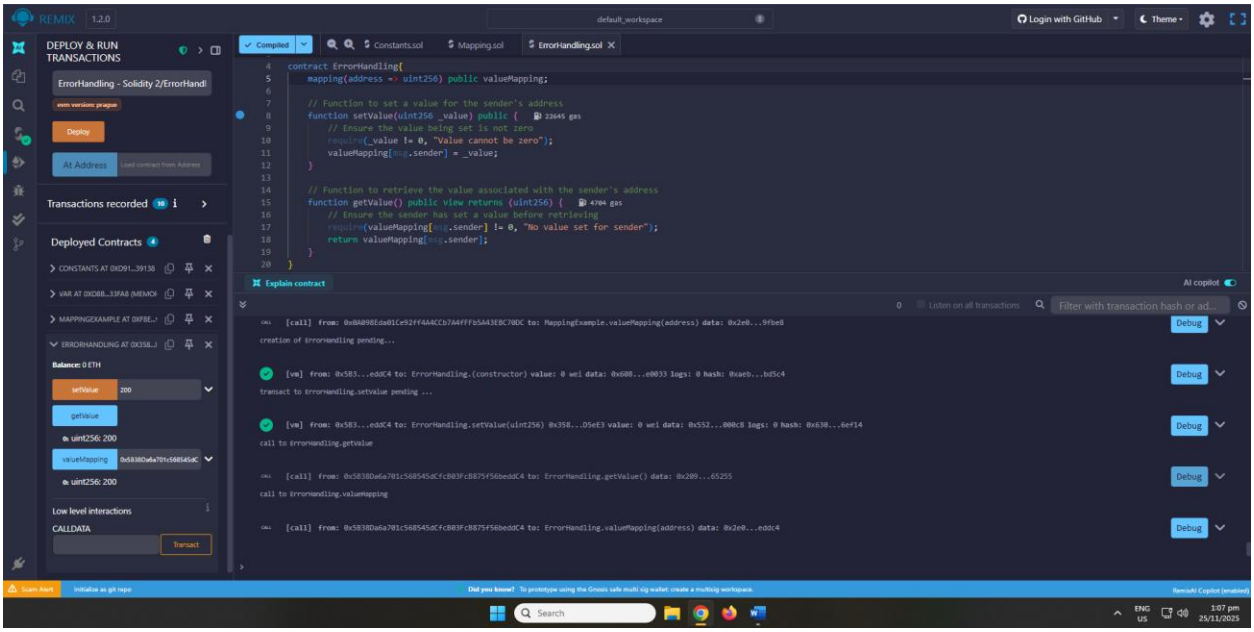




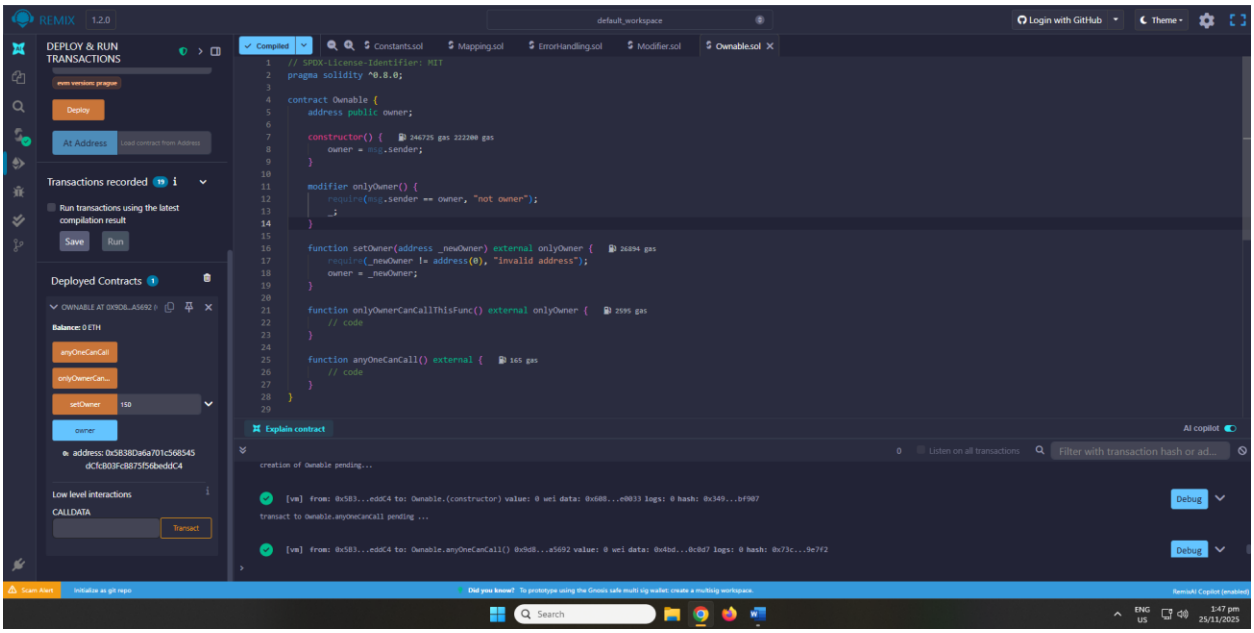
Mapping:



Error Handling:



Ownable Demo:



REFLECTION QUESTIONS

- **Function Modifiers & Ownable**
  When should you use a modifier like onlyOwner instead of inline checks, and what risks arise if ownership isn't managed properly?

  - Inline checks and modifiers has a similar form of validation control or ownership, but modifiers is more flexible in terms of an open construction of data rather than inline check could help you alone for contract size for particular scenarios.

- **Error Handling**
  How do you choose between require, revert, and assert, and why might custom errors be better than error strings?

  - Custom errors are more helpful because of how you could be more helpful in giving out the proper specific information regarding to an error that may occur. Nonetheless, using the advantage of every behavior of require, revert, and assert are one of the tools that could help the user to know the very factor of an error in the code.

- **Constants & Variables**
  When should a value be constant, immutable, or mutable, and how does that choice affect gas cost and flexibility?

- It is a scale system where you could choose for more flexibility but costly or a hard code that is not costly of your gas. Depending on your usage or block of code, you could use every advantage you could do in order to make your system more manageable and efficient in all its aspects. Therefore, being able to properly asses on the right approach that you could take is like a form of satisfaction for every users that you could have, in order for them to be in a comfortable approach of your system.