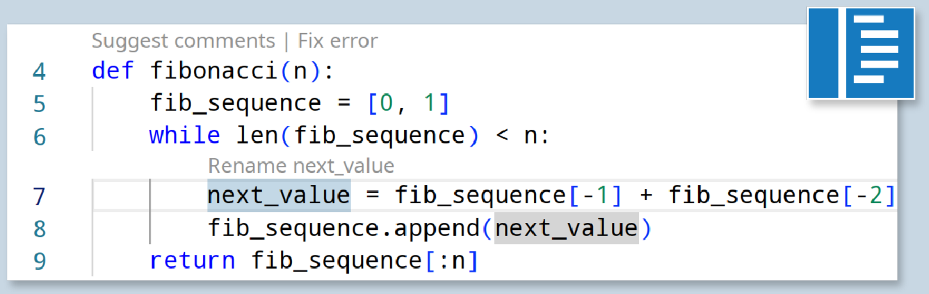
CodeImprove

A Code-OSS editor extension for enhancing code quality and readability of Python 3 code.

The client extension annotates code with options to perform each respective function directly in the editor.

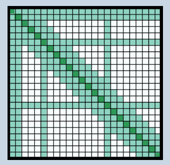
Clicking the option results in a request to the server being sent.



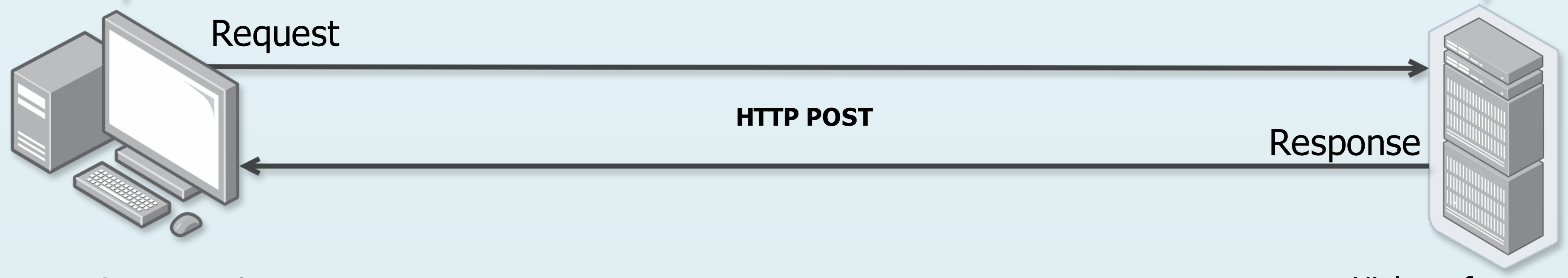
```
Suggest comments | Fix error
def fibonacci(n):
    fib_sequence = [0, 1]
    while len(fib_sequence) < n:
        Rename next_value
        next_value = fib_sequence[-1] + fib_sequence[-2]
        fib_sequence.append(next_value)
    return fib_sequence[:n]
```

The server is implemented in Python with the help of Pytorch and other libraries

The **Longformer** is a sparse attention Transformer architecture used to implement **Variable Renaming** and **Comment Suggestion**, achieving linear memory efficiency with respect to the growth of the sequence size



Error fixing leverages 4-bit quantization via **QLoRA** to achieve high memory efficiency while using a traditional model

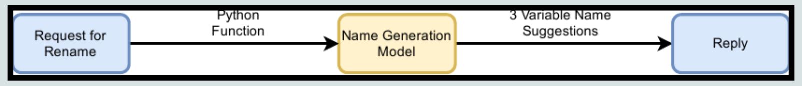


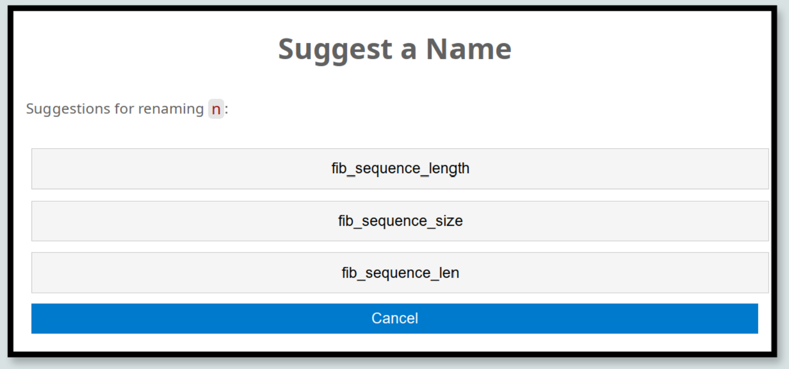
Low-performence client

High-performence server

Variable renaming

Generates three possible alternatives for the selected variable based on the context of the function.





Suggest a Name

Suggestions for renaming `n`:

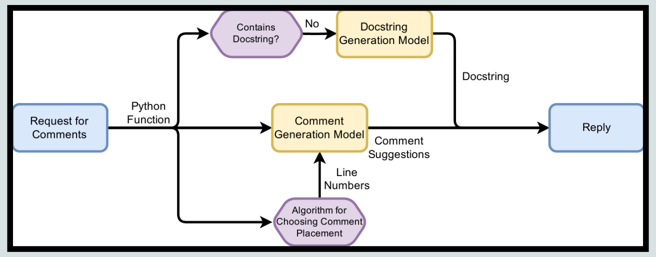
fib_sequence_length

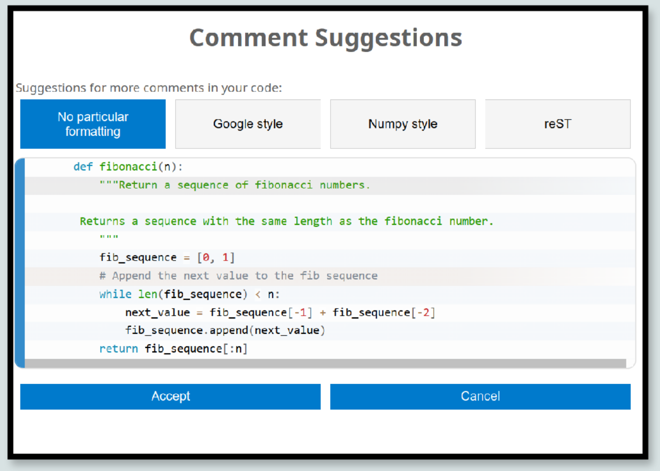
fib_sequence_size

fib_sequence_len

Comment Suggestion

Algorithmically chooses places to insert additional comments. It then generates those comments and the docstring, allowing the user to choose their preferred convention





Comment Suggestions

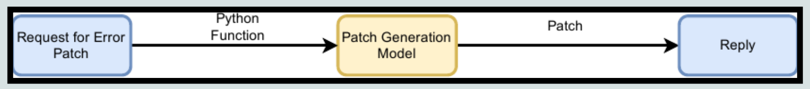
Suggestions for more comments in your code:

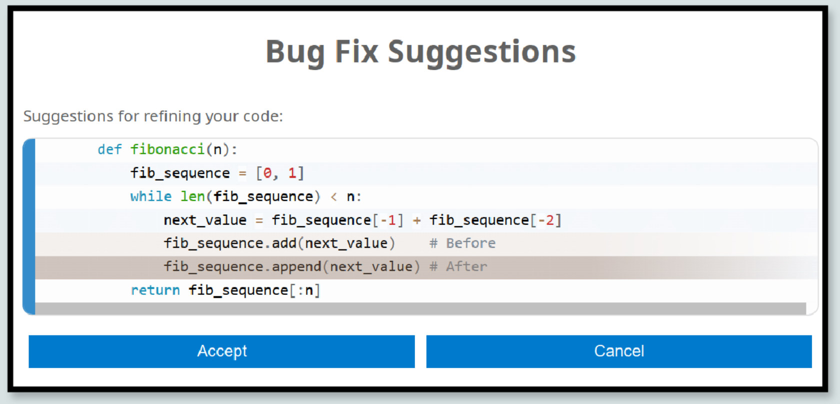
```
def fibonacci(n):
    """Return a sequence of fibonacci numbers.

 Returns a sequence with the same length as the fibonacci number.
    """
    fib_sequence = [0, 1]
    # Append the next value to the fib sequence
    while len(fib_sequence) < n:
        next_value = fib_sequence[-1] + fib_sequence[-2]
        fib_sequence.append(next_value)
    return fib_sequence[:n]
```

Error Fixing

Inspect the function for potential errors. If any are found, a patch with a fix is generated.





Bug Fix Suggestions

Suggestions for refining your code:

```
def fibonacci(n):
    fib_sequence = [0, 1]
    while len(fib_sequence) < n:
        next_value = fib_sequence[-1] + fib_sequence[-2]
        fib_sequence.add(next_value)    # Before
        fib_sequence.append(next_value) # After
    return fib_sequence[:n]
```

Generating code from textual description of functionality a Batchelor's thesis by **Ondřej Zobal**

Brno University of Technology, 2024