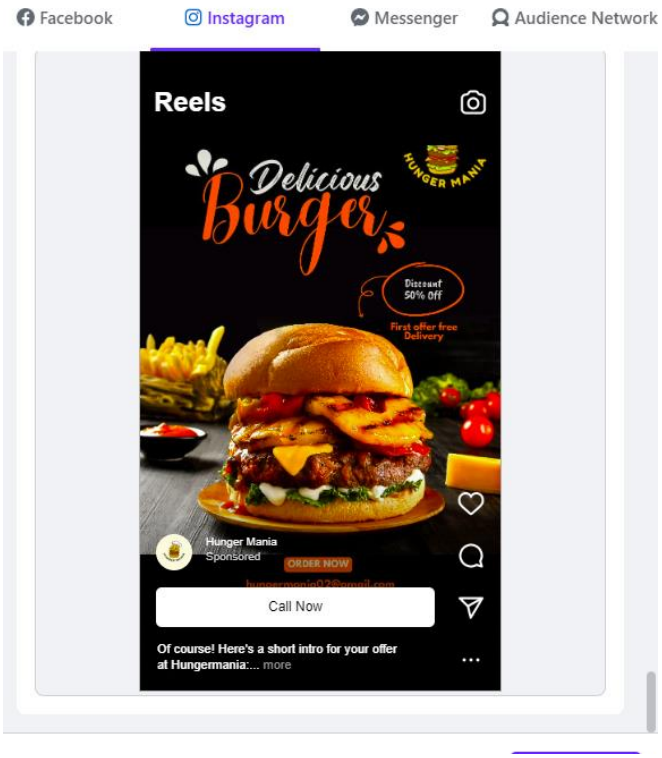
OUR CODE LAYOUT AND READABILITY:



1. **Consistent and Clear Indentation:** The project's codebase maintains a consistent and clear indentation style, making it easy to follow the code's structure and flow.
2. **Well-Structured Code:** The code is organized into logical sections and functions with meaningful names, making it straightforward to understand the purpose of each component.
3. **Descriptive Variable Names:** Meaningful and descriptive variable names are used throughout the project, enhancing code readability and reducing the need for constant reference.
4. **Modular Design:** The project follows a modular design, with well-defined functions and classes. This modularity allows for easy reuse of code components in various parts of the project.
5. **Minimized Nesting:** Deeply nested control structures have been minimized through the use of early returns and guard clauses, ensuring code remains concise and easy to follow.
6. **Proper Comments:** The code includes clear, concise, and up-to-date comments to explain complex or non-obvious sections, making it accessible for other team members and future maintainers.
7. **Docstrings for Documentation:** Functions and classes are documented with informative docstrings, providing detailed documentation about their usage, parameters, and return values.

8. **Readable Error Handling:** The project features user-friendly error handling, ensuring that error messages are clear and informative, simplifying debugging processes.
9. **Maintaining a Consistent Coding Style:** The project adheres to a consistent coding style and formatting guidelines, promoting readability and making it easy to integrate with other codebases.
10. **Structured and Readable Testing:** Unit tests are thoughtfully structured and written to ensure the correctness of functions and classes. This approach contributes to maintainable and readable testing code.
11. **Version Control Practices:** Version control systems are used effectively, with meaningful commit messages, branches, and clear documentation of changes. This aids in tracking code evolution and simplifies collaboration.
12. **Consistency Across the Codebase:** A uniform and consistent approach to code layout and readability is maintained across the entire codebase, ensuring that all parts of the project are equally well-structured.