## **Standard Operating Procedure (SOP)**

## **For**

## **GitHub Copilot Integration & Uses**

**1. Introduction**

GitHub Copilot is an AI-powered code completion tool that helps developers write code faster and with fewer errors. This SOP outlines the steps to set up and use GitHub Copilot with Visual Studio Code, JetBrains IDEs, and other tools.

**2. Prerequisites**

* A valid GitHub account with Calsoft email id.
* GitHub Copilot subscription with help of Calsoft’s IT team.
* Installed versions of Visual Studio Code, PyCharm, or other supported IDEs.

**3. Setup and Configuration**

**3.1 Visual Studio Code (VS Code)**

1. **Install GitHub Copilot Extension**:
   * Open Visual Studio Code.
   * Go to the Extensions view by clicking on the Extensions icon in the Activity Bar on the side of the window.
   * Search for "GitHub Copilot" and click "Install".
2. **Sign In to GitHub**:
   * After installation, you will be prompted to sign in to your GitHub account.
   * Follow the on-screen instructions to authorize GitHub Copilot.
3. **Configure Settings**:
   * Go to File > Preferences > Settings.
   * Search for "Copilot" to adjust settings such as enabling/disabling suggestions, configuring keybindings, etc.
4. **Using GitHub Copilot**:
   * Start typing in the editor, and GitHub Copilot will provide code suggestions.
   * Accept suggestions by pressing Tab or Enter.

For More Information Please visit - <https://code.visualstudio.com/docs/copilot/setup>

**3.2 JetBrains IDEs (e.g., IntelliJ IDEA, WebStorm, PyCharm)**

1. **Install GitHub Copilot Plugin:**
   * Open your JetBrains IDE (e.g., IntelliJ IDEA, WebStorm, PyCharm ) which You can download & install from [JetBrains IDEs](https://www.jetbrains.com/products/).
   * Navigate to File > Settings > Plugins.
   * Search for "GitHub Copilot" in the [Marketplace](https://plugins.jetbrains.com/plugin/17718-github-copilot) and click "Install".
   * Restart the IDE to activate the plugin.
2. **Sign In to GitHub:**
   * Go to File > Settings > Tools > GitHub Copilot.
   * Sign in with your GitHub account to link the plugin.
3. **Configure Settings:**
   * Adjust settings according to your preferences, such as enabling/disabling suggestions.
4. **Using GitHub Copilot:**
   * Start coding in your preferred language, and GitHub Copilot will suggest code completions.
   * Accept suggestions by pressing Tab or Enter

For more details [Please See](https://docs.github.com/en/copilot/using-github-copilot/getting-code-suggestions-in-your-ide-with-github-copilot?tool=jetbrains).

**3.3 Xcode**

1. **Install GitHub Copilot Extension:**
   * Ensure you have Xcode version 8.0 or above and macOS Monterey (12.0) or above installed.
   * Download the latest version of the GitHub Copilot for Xcode extension from the GitHub repository
   * Install the extension following the provided instructions.
2. **Sign In to GitHub:**
   * Open Xcode and go to Preferences > Extensions.
   * Enable the GitHub Copilot extension and sign in with your GitHub account.
3. **Configure Settings:**
   * Adjust settings in Xcode to enable or disable suggestions and configure keybindings.
4. **Using GitHub Copilot:**
   * Start typing in your Swift or Objective-C files, and GitHub Copilot will provide code suggestions.
   * Accept suggestions by pressing Tab or Option + Tab for multiline suggestions

For More details [please see](https://docs.github.com/en/copilot/using-github-copilot/getting-code-suggestions-in-your-ide-with-github-copilot?tool=xcode).

**4. Best Practices**

**1. Contextual Awareness**

* **Structured Code:** Ensure your code is well-structured and follows best practices. This helps GitHub Copilot understand the context better and provide more accurate suggestions.
* **Clear Comments:** Use clear and descriptive comments in your code. This can guide GitHub Copilot to generate suggestions that align with your intentions.

**2. Feedback Loop**

* **Provide Feedback:** Regularly provide feedback on the suggestions you receive. This helps improve the model's accuracy over time.
* **Accept or Reject Suggestions:** Actively accept or reject suggestions based on their relevance. This interaction helps refine the tool's future recommendations.

**3. Documentation**

* **Utilize Official Documentation:** Make use of the official GitHub Copilot documentation for advanced features, tips, and troubleshooting. This can help you get the most out of the tool.
* **Stay Updated:** Keep an eye on updates and new features released by GitHub Copilot. Staying informed can help you leverage new capabilities effectively.

**4. Code Quality**

* **Review Suggestions:** Always review the code suggestions provided by GitHub Copilot. Ensure they meet your coding standards and do not introduce errors.
* **Refactor When Necessary:** Use the suggestions as a starting point and refactor the code if needed to fit your project's style and requirements.

**5. Security and Privacy**

* **Sensitive Information:** Avoid using GitHub Copilot to generate code that handles sensitive information directly. Always review and sanitize such code manually.
* **Compliance:** Ensure that the code generated complies with Calsoft’s security policies including Calsoft Policy for Use of AI Coding Assistants and coding standards.
* **Documentation**: Utilize the official GitHub Copilot documentation for advanced features and troubleshooting tips. [See more](https://docs.github.com/en/copilot/using-github-copilot/best-practices-for-using-github-copilot)

**5. Troubleshooting**

* **No Suggestions**: Check if the extension/plugin is enabled and that you are logged into your GitHub account.
* **Performance Issues**: Ensure your IDE is updated to the latest version and that your system meets the necessary requirements.

**6. Additional Tools**

GitHub Copilot can also be integrated with other tools and IDEs. Refer to the [official GitHub Copilot documentation](https://docs.github.com/en/copilot/using-github-copilot/getting-code-suggestions-in-your-ide-with-github-copilot?tool=xcode) for detailed setup instructions for other environments.