Guia de Desenvolvimento: GitHub Desktop & Azure

1. Configuração Inicial (Clone)

Ao baixar o software, ir em File > Options na aba Git, preencher com:

Name -> Nome do desenvolvedor
Email -> O Lider do desenvolvimento deverá passar

Lembrando que esse e-mail não é próprio, o mesmo é usado em comum entre os desenvolvedores.

image.png
Image not found or type unknown

Ao baixar o projeto pela primeira vez, siga estes passos para manter a estrutura de pastas padronizada:

1. No **GitHub Desktop**, vá em `File` > `Clone Repository`.
2. Selecione a aba **URL**.
3. **URL do Repositório:** > `https://SuperSoftDelphi@dev.azure.com/SuperSoftDelphi/SistemaERPFISCO/_git/SistemaERPFISCO`
4. **Local Path (Atenção):** O GitHub Desktop tentará sugerir o nome `SistemaERPFISCO`.
   - **Obrigatório:** Altere o final do caminho para que a pasta se chame **`Codigo Fonte`**.
   - *Exemplo:* `C:\Codigo Fonte`

---

2. Fluxo de Branches

Trabalhamos com uma branch principal (`main`) e **branches de versão** para cada Sprint.

Iniciando uma Tarefa (Caso)

1. Certifique-se de que a **Current Branch** selecionada é a branch da versão atual (ex: `Branch 166`).
2. Clique em **Fetch Origin** para baixar as últimas atualizações.
3. Vá em **New Branch**.

4. Nomeie sua branch com o número do Desenvolvedor/Numero do Caso (ex: `vinicius.pereira_0053315`).
5. Garanta que a base da nova branch seja a branch da versão (ex: `166-Caso-0053542`).

---

3. Commits e Squash

Para manter o histórico do Azure DevOps limpo e fácil de ler:

- **Frequência:** Faça commits conforme avança no código.
- **Squash:** Antes de finalizar a tarefa, você deve consolidar seus commits.
  - No GitHub Desktop, vá na aba **History**.
  - Selecione seus commits da tarefa, clique com o botão direito e escolha **Squash commits**.
  - Isso transformará vários commits pequenos em um único commit robusto com a descrição da tarefa.

Exemplo de Commit

image.png
Image not found or type unknown

---

4. Atualização de Código (Rebase)

NUNCA utilize "Merge" para atualizar sua branch com a versão. Use sempre o **REBASE**.

O Rebase mantém o histórico linear, colocando seus commits no topo das últimas alterações da equipe.
Como fazer:

1. Com sua branch de tarefa selecionada, vá no menu superior: `Branch` > **Rebase Current Branch...**
2. Selecione a branch da versão (ex: `166-Caso-0053542`).
3. O GitHub Desktop processará a reescrita do histórico. Se houver conflitos, resolva-os no VS Code/Delphi e finalize o rebase no GitHub Desktop.

OBS:

Caso apareça uma mensagem de erro ao clonar, dar fetch.

Só clicar em Clone > Generate Git Credencials e copiar o Password.

Clipboard - 23 de fevereiro de 2026 às 15_05.png
Image not found or type unknown

Regras de Ouro (Resumo)

O que fazer	O que NÃO fazer
Usar a pasta **Codigo Fonte**	Deixar o nome padrão `SistemaERPFISCO`
Fazer **Rebase** da versão	Dar "Merge" da versão na sua branch
Fazer **Squash** dos commits	Enviar dezenas de commits de "ajuste" no PR
Criar branch a partir da **Versão**	Criar branch a partir da `main` sem orientação

---

Revision #1
Created 24 February 2026 16:49:44 by Vinicius Pereira
Updated 24 February 2026 16:50:27 by Vinicius Pereira