

# Resolving Self-Induced Git Conflicts

This tutorial walks through how to resolve merge conflicts in Git when multiple team members make overlapping changes, and when a pull request (PR) was never opened before others merged their work.

## ✍ Scenario

- I made edits (`README.md`, deleted `new 1.txt`) and committed to my feature branch.
- I forgot to open a pull request.
- Later, a teammate updated the same `README.md`, opened a PR, and merged it into `main`.
- Now my branch conflicts with `main`.

## ⚡ Step 1: Fetch the latest changes

Update my local repo with the remote state, without overwriting local work:

```
git fetch origin
```

This tells Git what has changed on GitHub, including new commits and branches.

## ⚡ Step 2: Merge `main` into your branch

While on my feature branch (e.g., `carrie-docs`):

```
git merge origin/main
```

This attempts to combine my branch with the latest `main`. If there are overlapping edits, Git will flag conflicts.

## 📸 Screenshot: Terminal showing README.md conflict markers

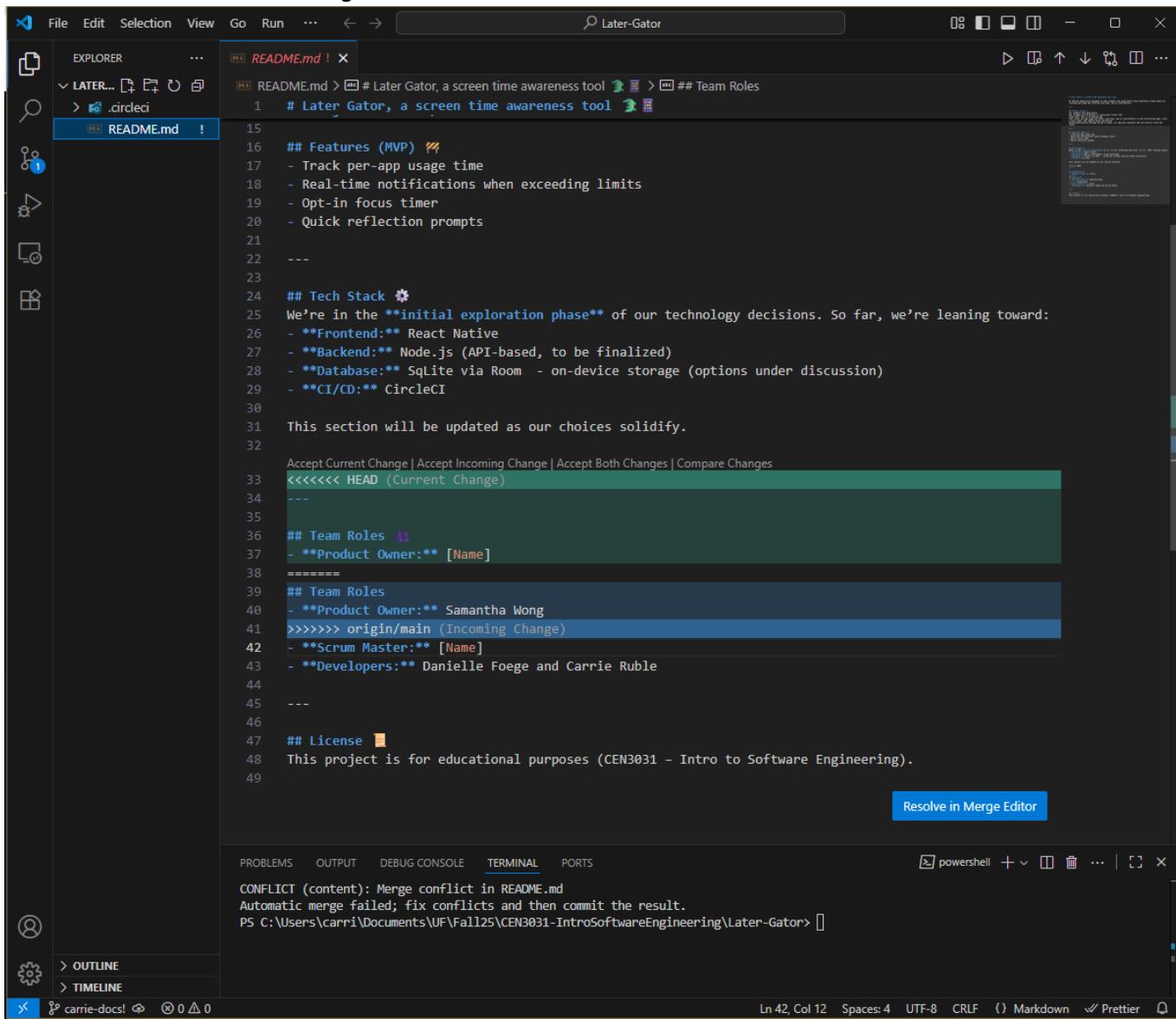
```
PS C:\Users\carri\Documents\UF\Fall25\CEN3031-IntroSoftwareEngineering\Later-Gator> git fetch origin
remote: Enumerating objects: 6, done.
remote: Counting objects: 100% (6/6), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 4 (delta 2), reused 2 (delta 1), pack-reused 0 (from 0)
Unpacking objects: 100% (4/4), 1.11 KiB | 47.00 KiB/s, done.
From https://github.com/swong36/Later-Gator
  182947e..e2e6b1f  main      -> origin/main
* [new branch]      sam-PM    -> origin/sam-PM
② PS C:\Users\carri\Documents\UF\Fall25\CEN3031-IntroSoftwareEngineering\Later-Gator> git merge origin/main
Auto-merging README.md
CONFLICT (content): Merge conflict in README.md
Automatic merge failed; fix conflicts and then commit the result.
③ PS C:\Users\carri\Documents\UF\Fall25\CEN3031-IntroSoftwareEngineering\Later-Gator>
```

## ⚡ Step 3: Identify the conflict

Git will mark the conflicted file(s) in red. Opening them in VSCode shows conflict markers:

```
<<<<< HEAD
(my version)
=====
(teammate's version)
>>>>> origin/main
```

📸 Screenshot: VSCode showing README.md conflict markers

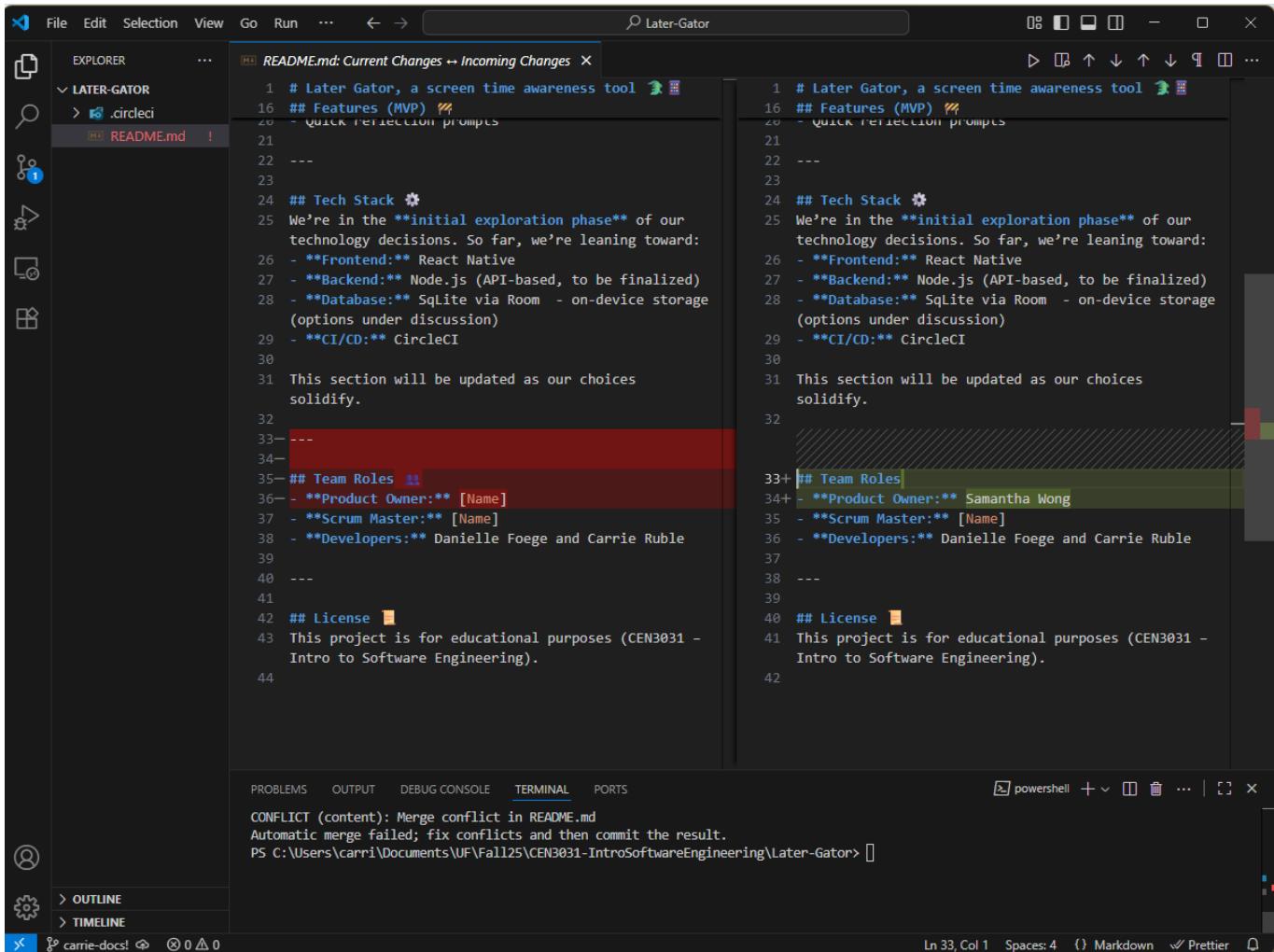


## ⚡ Step 4: Open the merge editor

Click “**Resolve in merge editor**” in VSCode. I see side-by-side comparisons:

- **Current changes** = my branch
- **Incoming changes** = **origin/main**
- **Result** = the merged file preview

## Screenshot: VSCode merge editor UI



## ⚡ Step 5: Accept and clean up

- Use “**Accept Both Changes**” when appropriate.
- Then manually delete duplicate headers or redundant lines.

Example resolution:

```

## Team Roles 🤝
- **Product Owner:** Samantha Wong
- **Scrum Master:** [Name]
- **Developers:** Danielle Foege and Carrie Ruble

```

## Screenshot: Cleaned up README.md in VSCode

The screenshot shows a Microsoft Visual Studio Code (VSCode) interface with a dark theme. The main editor tab is titled "README.md". The code content is as follows:

```
1 # Later Gator, a screen time awareness tool 🐊
2
3 ## Features (MVP) 🎯
4 - Track per-app usage time
5 - Real-time notifications when exceeding limits
6 - Opt-in focus timer
7 - Quick reflection prompts
8
9 ---
10
11 ## Tech Stack ⚙️
12 We're in the **initial exploration phase** of our technology decisions. So far, we're leaning toward:
13 - **Frontend:** React Native
14 - **Backend:** Node.js (API-based, to be finalized)
15 - **Database:** SQLite via Room - on-device storage (options under discussion)
16 - **CI/CD:** CircleCI
17
18 This section will be updated as our choices solidify.
19
20 ---
21
22 ## Team Roles 📜
23 - **Product Owner:** [Name] ...
24 ## Team Roles
25 - **Product Owner:** Samantha Wong
26 - **Scrum Master:** [Name]
27 - **Developers:** Danielle Foege and Carrie Ruble
28
29 ---
30
31 ## License 📄
32 This project is for educational purposes (CEN3031 - Intro to Software Engineering) Resolve in Merge Editor
33
34
35
36
37
38
39
40
41
42
43
44
45
46
```

The code editor has a blue box highlighting the line "This project is for educational purposes (CEN3031 - Intro to Software Engineering)". Below the editor, the "PROBLEMS" panel shows a single conflict entry:

CONFLICT (content): Merge conflict in README.md  
Automatic merge failed; fix conflicts and then commit the result.  
PS C:\Users\carri\Documents\UF\Fall25\CEN3031-IntroSoftwareEngineering\Later-Gator> []

The bottom status bar shows: Ln 35, Col 16 (46 selected) Spaces: 4 UTF-8 CRLF {} Markdown ✅ Prettier

## ⌚ Screenshot: Another look at cleaned up README.md in VSCode

```

1 # Screen Time Awareness Tool
18 ## Tech Stack
23
24 ## Development Setup
25 1. Clone the repository
26     ```bash
27     git clone https://github.com/<org>/<repo>.git
28     cd <repo>
29     ```
30
31 2. Install dependencies
32     ```bash
33     npm install
34     ```
35 3. Run the app in development
36     ```bash
37     npm start
38     ```

39 ## Team Roles
40 - **Product Owner:** Samantha Wong
41 - **Scrum Master:** [Name]
42 - **Developers:** Danielle Foege
43

Result README.md
1 # Later Gator, a screen time awareness tool
24 ## Tech Stack
36
37 Incoming + Current | Remove Incoming | Remove Current
38 This section will be updated as our choices solidify.
39
40 ## Team Roles
41 - **Product Owner:** Samantha Wong
42 - **Scrum Master:** [Name]
43 - **Developers:** Danielle Foege and Carrie Ruble
44

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
CONFLICT (content): Merge conflict in README.md
Automatic merge failed; fix conflicts and then commit the result.
PS C:\Users\carri\Documents\UF\Fall125\CEN3031-IntroSoftwareEngineering\Later-Gator>

```

## ⚡ Step 6: Complete the merge

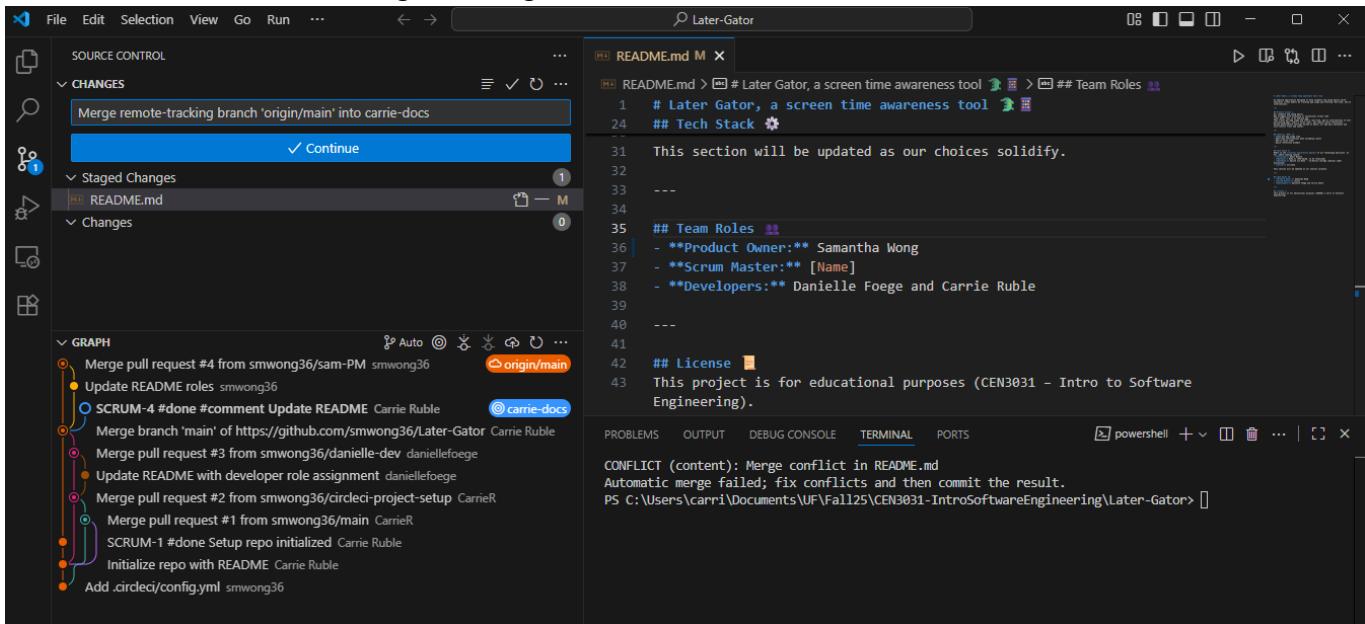
Once the file looks correct:

```

git add README.md
git commit -m "Resolve merge conflict in README.md"
git push origin carrie-docs

```

## 📸 Screenshot: VSCode Showing that I'm good to commit



## ⚡ Step 7: Open my pull request

Back on GitHub, I now see:

- My conflict resolution commit.
- My previous commit (e.g., SCRUM-4).

Open a PR, and GitHub should report “**No conflicts with the base branch.**”

## 📸 Screenshot: GitHub PR view after resolving conflicts

**Update README Remove 'new 1.txt' #5**

CouldBeYourMom wants to merge 2 commits into `main` from `carrie-docs`

**Conversation** 0    **Commits** 2    **Checks** 0    **Files changed** 2    +22 -33

**CouldBeYourMom commented now**  
No description provided.

**CouldBeYourMom added 2 commits 16 hours ago**

- SCRUM-4 #done #comment Update README ... d608d83
- Resolve merge conflict in README ... 0326e06

**All checks have passed**  
1 successful check

**No conflicts with base branch**  
Merging can be performed automatically.

**Merge pull request**

Reviewers: daniellefoege (Request)  
Suggestions: None yet  
Assignees: None yet  
Labels: None yet  
Projects: None yet  
Milestone: None yet  
Development: Successfully merging this pull request may close these issues.

## ✓ Outcome

- Both sets of changes (mine + your teammate's) are now in `main`.
- History clearly shows the conflict and how it was resolved.
- This is a perfect example of why:
  - PRs must be opened promptly.
  - Teammates should review open PRs before merging their own work.