

## 6.5.7 Save Your Progress with Git

Congratulations on completing the application! We've finished the work on this GitHub issue, so let's revisit the Git branch workflow:

1. Verify that the correct files were modified with `git status`.
2. Stage and commit your changes with `git add` and `git commit`.
3. Push the branch to GitHub with `git push origin feature/languages`.
4. Switch branches with `git checkout develop`.
5. Merge the new feature into the develop branch with `git merge feature/languages`.
6. Push the updated develop branch to GitHub with `git push origin develop`.

Finally, navigate to GitHub and close the corresponding GitHub issue!

DEEP DIVE ▲

## DEEP DIVE

---

Instead of manually closing the issue, you can also tell GitHub to close it automatically. You can do this by adding the following phrase to your commit message: `fix #issueNumber`, where `issueNumber` is the number of the GitHub issue you'd like to close.

To learn more, read the [GitHub docs on closing issues with keywords](https://help.github.com/en/github/managing-your-work-on-github/closing-issues-using-keywords) [\\_ \(https://help.github.com/en/github/managing-your-work-on-github/closing-issues-using-keywords\)\\_](https://help.github.com/en/github/managing-your-work-on-github/closing-issues-using-keywords).

Because this is the last issue to be closed, don't forget to merge the `develop` branch into the `master` branch. We've already deployed the application; we just need to get the updated code into the `master` branch!

Finally, review this lesson's takeaways with the following knowledge check: