

GitHub Version Control Process

This document outlines the Git workflow and branching strategy used in this repository. It defines how developers should work with branches, create pull requests, and manage updates to both development and production environments.

Branching Strategy

- **main** → Production branch
- **development** → Staging/pre-production branch
- **Feature branches** → Created from **development** for all work

Branch	Purpose	How to update
main	Production releases	Pull Request from development
development	Active development and testing	Pull Request from feature branches

Creating a New Branch

All feature or bugfix branches must be created from **development**.

```
git checkout development
git pull origin development

git checkout -b feature/add-new-model
```

Then push your new branch:

```
git push -u origin feature/add-new-model
```

Creating a Pull Request

Feature → Development

Once your feature is complete:

1. Push your branch to GitHub
 2. Open a Pull Request **into development**
 3. Request a review
 4. Merge the PR using the GitHub interface
-

Development → Main

When ready to release to production:

1. Open a Pull Request **from development into main**
2. Ensure CI passes and reviewers approve
3. Merge the PR using the GitHub interface

Branch Protection Rules

Branch protection is configured in GitHub to enforce this workflow:

- Pull requests are required for all updates
- Reviews are required before merging
- CI/CD checks must pass
- Direct pushes are blocked by configuration

Example Workflow

```
[feature/add-new-model]
  |
  v
[development] <-- PR from feature branch
  |
  v
[main] <-- PR from development (for release)
```

Related Documents

- [development_process](#)
- [installation_and_setup](#)
- [ci_cd_workflows](#)