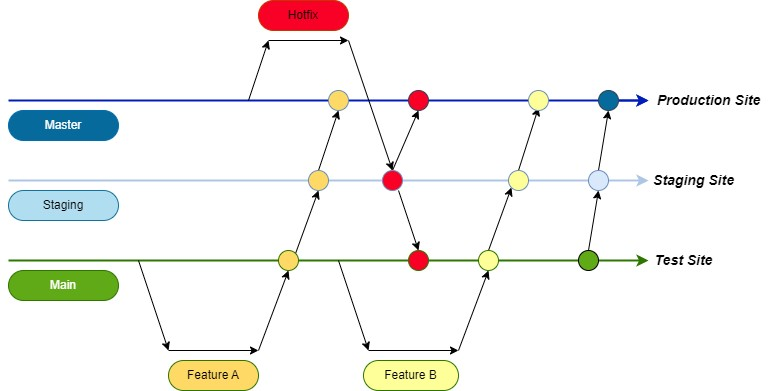
**Branching strategy**

This is the fundamental schematic of the branching strategy used in all projects.



Here's a quick explanation of what it means.

We require the following three branch as a minimum.

1. Main
2. Staging
3. Master

**Main Branch**:-

* This is the development's primary branch.
* It provides with a test environment. QA will do the initial round of testing.
* The developer will create a new branch from main branch with the ticket number and heading whenever we begin a new development.
* The developer will move the code into the main branch after finishing the development.

**Staging:**

* This is the preliminary production environment.
* Only when all testing has been completed in the test environment, we can push the main branch's code to the staging branch.
* Rather than combining(merging), we must individually push(cherry-pick) each commit that we require to the staging branch. This will limit the migration of untested/unwanted code to staging.
* Here, functionality testing is done by QA.
* The production data copy is present.
* After QA is completed, client/user testing will take over.

**Master**:

* This is the production branch where the production site is located. We will merge the code from the staging branch to the master branch as soon as the client or a higher authority approves the project.
* We occasionally only get a few tickets approved. At that point, we will do the cherry pick(required commits) to the newly generated branch from master branch.
* Finally we will merge the code to master.

**Hot fix**:

* This is the branch that is established whenever an urgent code move to production or production fix is required.
* We shall make a new hotfix branch from the master in this instance. The developer has finished working on this branch.
* This code has to be tested in a staging environment. After QA testing is complete, the code will be pushed to the production environment and moved to main branch simultaneously.