**GITHUB**

**Create New**

* **Import repositary:** Creating a copy of existing repository without affecting the original one. Also known as ***forking***.
* **Codespace:** Uses ***github.dev*** to **edit** codes online on GitHub itself.
* **Gist:** Small piece of code created to be instantly **share** or **integrate** with the project.
* **Organization:** Created for enabling team co-operative features of GitHub.

**Issues**

* It is a feature allowing **structured communication** among developers.
* Developers can share update, bug fixing, milestone, task etc information.

**Pull Requests**

* Meaning **request** for making **changes** in the code.
* Even **code reviews** can be written there for it.
* Codes can be approved and merged into main branch if owner agrees.
* **Workflow:** Automated script to handle jobs.
* **Force push:** Rewriting history of repository.

**Branching**

* **Main branch:** Represents latest **stable** version of the code.
* **Feature branch:** Used for adding new feature to the code.
* **Branch switching:** Developers can work on independent branches **without** affecting the main branch.
* **Merging:** Finally used after testing.
* **Deleting branch:** Deletion of **unwanted** branches.

**Top Bar**

* **Issues:** A built-in **todo list** by GitHub.
* **Actions:** Various built-in **support systems** for deployment, security, automation etc as per the programming language & project’s nature.
* **Security:** Provides user with various options to take **security measures** for the project.

**Sections Under Security**

* Security policy
* Security advisories
* Private vulnerability reporting
* Dependabot alert
* Code scanning alerts
* Secret scanning alert

**Sections Under Insights**

* **Pulse:** Shows statistics about **pulls** and **merges** etc.
* Contributor
* Community
* **Community standards:** GitHub provided checklist to complete and fit in a community.
* **Traffic:** Statistics about views and visitors.
* Commits
* Code frequency
* **Dependency graph:** Shows statistics related to tools or languages you are relying to
* Network
* Forks

**Settings**

General:-

* **Features:** Can create discussion space, sponsorship option etc.
* **Danger zone:** Direct changes to **nature** of repository like making it public to private.

Access:-

* **Collaborators:** Collaborators can access the repository and make changes in it.

Code and automation:-

* Its more of **security aspect** of your repo.
* **Branches:** Create branch protection rules.
* **Rules:** Make your own rules, but for private repos it can be done only with GitHub Team.
* **Action:** Some **more** security measures that could be taken.
* **Webhooks:** URLs to send notification about changes in your repo.