

Level up with GitHub actions, bots and CI/CD

Manaswini Das

Set up CI/CD

- Development approaches
- Integrate/Deploy more often to alleviate surprises and challenges during production
- CD is a step further
- Thrusts automation and reduces human intervention

Example: <https://github.com/manaswinidas/kogito-docs>

Reference: <https://www.growingwiththeweb.com/2016/07/enabling-pull-requests-on-github-wikis.html>

Building GitHub actions

GitHub Actions (as for other CI/CD services/platforms):

- Triggers on events
- Prepare environment(s)
- Runs workflow(s)
- Report results

Types of actions

- **Dockerfile** action:
 - Environment defined by **container layers**
 - Workflow defined by entry point bash script
- **JavaScript** action:
 - Environment defined in the **package.json**, as NPM modules
 - Workflow is defined by the JavaScript/TypeScript code

Show time !

References:

<https://notiz.dev/blog/build-and-publish-your-first-github-action>

<https://medium.com/@michaelekpang/creating-a-ci-cd-pipeline-using-github-actions-b65bb248edfe>

Building GitHub bots

- Apps that runs automation on GitHub
- Uses GitHub WebHooks and APIs
- Enables one to focus on real collaboration

References: <https://developer.github.com/v3/activity/events/types/> ,
<https://github-bot-tutorial.readthedocs.io/en/latest/>

Examples:

<https://github.com/TimonVS/epic-generator>,
<https://github.com/manaswinidas/github-bot-example>

“In real open source, you have the right to control your own destiny”

- Linus Torvalds

You can find me



dasmanaswini10@gmail.com



@manaswinidas



@ManaswiniDas4