OMP 2019 Internship Application - Build DockerHub development stacks

About Me

Name and Contact Info

Name: Pranjal Tale
University: Indian Institute of Technology, Roorkee
Major: Applied Mathematics
Email: pranjaltale16@gmail.com
Github: github.com/pranjaltale16
Personal Website: pranjaltale16.github.io
Time Zone: IST (UTC +5:30)

Programming Experience

Work experience

Chief Coordinator (Information Management Group)

- Encouraged the adoption of new fields and technologies in the group with a push towards Container Infrastructure, Open Source and Network Programming.
- Reformed the entire stack of IMG to run on all the future-ready cutting-edge technologies adding a modular structured approach and making them open source so that anyone can use them. Github: github.com/IMGIITRoorkee

Software Development Internship (Hackerrank)

- Extended Firepad to support collaborative editing for Monaco-editor.
- Developed a code evaluator which evaluates every submission in an isolated environment using **container** properties. The entire project is written in **Golang** to get high concurrency for sync and evaluation

Software Development Internship (Hackerrank)

- Developed a portable static binary in Golang to compile and run code of any language inside a container. Used **Golang** templates and env variables to develop the logic of binary. Binary handles timeout and logs profiling of the submissions.
- Developed runtime Images for multiple languages. Images can be found at https://hub.docker.com/u/coderunner
- Wrote **k8s** charts and wrote **Helm** charts for the same.

- **Summer of Haskell ‘17**, CodeWorld (github.com/google/codeworld)
  Developed a standalone executable Cabal package for source code compilation wrapped over GHCJS; added test suite for the same and regex parser for filtering the error messages. Made a custom tree parser using haskell-src-exts and syb, to check for syntax errors before compilation. Tech-Stack: Haskell, Shell scripting, SNAP, git
- **Software Development Internship (Vernacular.ai)**
  Developed a python package to parse text into structured data. It parses temporal expressions from different languages and recognize quantities described in different ways. Tech Stack: Python, Django, git

Contributions to open source

I’ve contributed in several organisations. These are the list of contributions in them.

Docker-yii2 (https://github.com/pranjaltale16/docker-yii2)

- Developed a dockerized version of yii2 advanced template.
- Used docker-compose to start and bundle the containers and load them in the same network, not exposing the internal ports of Database on host machine.

Firepad (https://github.com/firebase/firepad)

- Added support for Monaco editor in firepad (**Merged**). Pull Request.

CodeWorld (github.com/google/codeworld)

- Added haskell-src-exts support in codeworld-compiler **(Merged)**. #542.
- Fixed issue #505 **(Merged)**. #539.
- Separating error sanitizer logic to parse runtime errors **(Closed)**. #536.
- Update test cases for codeworld-compiler **(Merged)**. #510 and #520.
- Standalone compiler for CodeWorld **(Merged)**. #503.
- Shifted regex parser from JavaScript to Haskell **(Merged)**. #502.
- Corrected typo **(Merged)**. #496.
- Fixed issue #420 **(Merged)**. #479.

Cadasta (https://github.com/Cadasta/cadasta-platform)

- Add user profile images #110 **(Closed**). Pull Request.

Sympy (https://github.com/sympy/sympy)

- Correct the diophantine function in Sympy module. (**Merged**). Pull Request.

The Project

The Problem and Motivation

The main goal of the project is to produce a suite of Docker images for OpenSUSE and ClefOS. My primary focus for this project will be around producing Docker images available for OpenSUSE and developing a single mechanism for image modifications. If time persists I'll also work around automation of whole process and providing a test suite for the same. Making the whole process simplified and streamlined. As the part of internship, **I will focus on:**

1. Patch based image model
2. OpenSUSE based image suite
3. Test suite and CI Pipeline(Optional)

The overall task, their order, timeline, and subtasks are formulated by me in a rough format of what I personally figured out from the project description. We can modify or update the overall structure with respect to tasks and timeline, their priority order, and project goals after discussing with Mentor(s) and thus forming a solid outline of what should be done as a part of the project.

Detailed description

- ### Patch based image model.
  Designing and creating the okcd-images repo based model for other docker repository. In this, first task would be to find the seminal image(base), and if the seminal image is available, write s390 patch for it. This way we would be able to add the same model to other repos as available in okcd-images. After completion of this task, both https://github.com/nealef/okcd-images and https://github.com/nealef/docker will follow the same mechanism for image creation

- ### OpenSUSE based image suite.
  Producing the equivalent suite based on OpenSUSE. Once the patch based model is implemented we can extend the same approach for creating the OpenSUSE image suite. In this task, my primary role would be to check for the packages available for CentOS and update the docker images. I'll make sure to update README and make note of

packages and images which cannot ported because of some package unavailability.

- **Test suite and CI Pipeline**(Optional).

  Writing the test suite for images and create a pipeline to update the images on docker hub. If the above tasks doesn't take much time then I can definitely work on this. In this, first subtask would be to create a pipeline so that it will be easier to push images on hub, test them and make sure that Images generated are right at the same time. The next task would be of adding tests for images. I will make sure to add the CI pipeline and tests to the project after Internship, if I'm unable to put this in internship timeline.

Proposed Timeline (US Schedule)

Pre Community Bonding Period (March 17th - Apr 29th).

I will utilise this time to get familiarized with the codebase, Docker, available stack of images and packages. I'll also discuss with my mentor(s) and get the timeline and project goals reviewed, and thus forming a solid outline of what should be done as a part of the project.

Community Bonding Period (Apr 29th - May 17th).

During this period, I'll try to do more research on to the project and associated tasks, see for the different options available, read the documentation, fix small issues to get started and get to know my mentor(s) and other fellow community members so that we can work together with an amplified efficiency.

Week 1-3 (May 20th - June 10th)

Updating the docker images on s390 patch model and writing essential scripts required.

Week 4-5 (June 10th - June 24th)

Continuing the task of docker images modification and simultaneously updating the DockerHub. Regular PRs for TODOs, documentation etcetera.

Phase 1 evaluation submission

Week 6-7 (June 24th - July 8th)

Once the migration in done, following the same model I'll start writing Docker images based on OpenSUSE. Proper documentation and Notes for any additional requirement or non availability of image due to package deficiency.

Weeks 8-9 (July 8th - July 22nd)

Completing the OpenSUSE Docker images part and start working on CI Pipeline for automation and streamline flow of updates and push on DockerHub

Weeks 10 (July 22nd - July 28th)

Writing tests, improving documentation and submitting code samples.

Final evaluation

Notes

- After completing any subtask or before making any pull request I will make sure to update the documentation and run test suites. Also, I'll make sure to always have a WIP Pull Request, so that code can be reviewed regularly and progress can be measured easily
- I'll use Github-projects for task management and keeping the track of all the records, issue and tasks.
- I am going to use Community Bonding and Pre Community Bonding period to resolve small bugs, digging into the codebase, discussing, strategizing, planning and finalizing things with the mentor.
- I can devote 45-50 hours per week.
- I tried my level best to present maximum things and up to the best of my knowledge and further proposal can be modified according to the tasks priority and the project goals after discussion with mentor(s).
- Even after the Internship programme, I will continue contributing to various OMP projects that will keep me in regular touch with the community and I'll try my level best to improve it to the extent possible from my side.