Artwork by Ankit, published under CC0.

Vision

Today's world is driven by software. We are able to solve increasingly complex problems with only a few lines of code. However, with increasing complexity, code quality becomes an issue that needs to be dealt with to ensure that the software works as intended. Code reviews have become a popular tool to keep the quality up and problems solvable. They make out at least 30% of the amount of time spent on the development of a software product.

Static code analysis and code reviews are converging areas. Still, they are still treated seperately and thus their full synergetic potential remains unused. With GitMate, we want to reinvent the code review process. Our product will integrate static code analysis directly into the code review process to reduce the number of bugs while leaving more time for the development of your favorite features.

Our product, the interactive code review bot "GitMate", is not only an easily usable static code analyser, but also actively supports the development process without any overhead for the developer. GitMate is as easy to use and interact with as a collegue next door and unique in its capabilities to even fix bugs by itself. It thereby reduces the amount of work of the reviewer, allowing him to focus on semantic problems that cannot be solved automatically.

Product

GitMate is a code review bot. It uses coala [1] to perform static code analysis on GitHub Pull Requests [2]. It searches committed changes for possible problems and drops comments right in the GitHub review user interface, effectively following the same workflow of a human reviewer. Any software developer or reviewer can respond directly in the GitHub UI and tell GitMate to ignore or even fix the indicated problem.

This process merges into the existing workflow of companies using GitHub or GitHub Enterprise and therefore poses no overhead at all to developers. The simplicity and unique interactivity makes GitMate easier and faster to use than its competitors.

Our long term plans include support for GitLab, Stash, BitBucket, Crucible and other code review platforms, to widen the target audience. We also plan to provide a web UI that allows users to write and use arbitrary static code analysis routines to make custom checks and solutions possible.

Extensive documentation will be provided, as well as further support via E-Mail, private chat, a public Gitter [5] support channel and a public bug tracker.

Implementation

The coala project [1] provides a target language independent static code analysis framework and allows to plug in arbitrary analysis routines in a simple manner. Integrating with existing linters (e.g. PyLint) provides a large number of sophisticated static code analysis routines with little effort. coala is a nonprofit project and developed open source.

GitMate will be built as a hosted webservice which uses the GitHub API [3] to communicate with the user via the existing GitHub UI. The service will be executed on a PaaS or an IaaS. The initial product will be bare and simple: Users can give GitMate access to a repository, and coala will be run on every creation or change of a pull request. Occurring issues are posted on GitHub. A configuration file for coala can be created and submitted with the project. As the product evolves, we will add user interaction and more features as they are requested by customers.

In the long term, GitMate will evolve to assist the whole workflow by automatically performing actions upon requests from users, given via GitHub comments. GitMate will be able to automatically rebase and merge branches or generate releases.

The addition of support for GitLab, Stash, Bitbucket, Crucible and other review UIs is trivial at a later point in time, as all of them provide an API similar to the GitHub API. The execution of custom analysis routines written by the customer will not pose security issues when existing environment isolation methods like Docker [6] are used.

Team

GitMate will be lead by Lasse Schuirmann and implemented together with Fabian Neuschmidt, Mischa Krüger and Udayan Tandon. All four are also a core part of the international team that works on coala.

Lasse Schuirmann began to learn about computer science with an age of 14 when he implemented a new image processing algorithm. Since then, he passionately worked on many projects in his freetime, including an autonomously flying model plane and a minimal operating system kernel. Since 2012 Lasse is employed at Draeger Medical GmbH while being enrolled at the Hamburg University of

Technology (TUHH) for computer science and engineering. In 2013 he founded the coala project together with Fabian Neuschmidt. Lasse received the Google Summer of Code (GSoC) stipend in 2014 under the GNOME organization. He acquired valuable contacts as well as much experience in the development of Open Source projects through this program. In 2015, he grew into a GNOME Foundation member and became the main administrator of GSoC for GNOME while mentoring several students for the same program. While administrating 24 stipends he learned a lot about organizational and financial issues. He has written his Bachelor thesis about a code clone detection algorithm, implemented through coala. In late 2015 Lasse was able to establish a cooperation with the Technical University of Berlin to offer projects and theses in conjunction with coala in the future.

Fabian Neuschmidt, the co-founder of coala, has been involved in the project from the start and massively shaped and influenced the code architecture of coala through its iterations. He is a former scholarship holder of the German National Academic Foundation and enrolled at the TUHH for mechanical engineering. Through his engagement in several departments of the nonprofit organization e-gnition at the TUHH he has earned additional experiences with marketing, sponsor acquisition and the management of larger teams.

Mischa Krüger successfully developed a textual game engine as the leader of a small team under the mentorship of Lasse in 2012 within several weeks. Together with Lasse he decided to continue this project as free software. Over time he got more and more involved in the coala project and is a core maintainer since 2015.

Udayan Tandon is a student at IIIT Delhi, pursuing his degree in computer science while learning various web-backend technologies in his free time. In 2014 he worked on designing APIs for web and android applications until he was employed by EMC in 2015. In the same year, Udayan received the Google Summer of Code stipend for the development of a graphical user interface prototype for coala. He is now the maintainer of the coala UI which has grown to his main freetime project. Through diverse experiences in his freetime, university and his employment at EMC, Udayan gained insights in web APIs and backends making him an invaluable part of the team for deploying our webservice.

The combination of many theoretical and practical experiences from very different sources provides a great basis for our project and the knowledge needed to raise a successful business. All team members committed great efforts to bring coala to the current state, mostly in their free time. We are dedicated to improve the world of coding by taking effort from the shoulders of developers while supporting free software to establish good and open standards for static code analysis.

Target Audience

The target audience of the initial product consists of:

- Personal users and small nonprofit organizations using GitHub for open source software.
- Personal users using GitHub for developing private (proprietary) software.

- Companies using GitHub or GitHub Enterprise for developing commercial software.

The target audience will be extended as GitMate gains support for other review user interfaces.

Existing Similar Solutions

There are a number of products on the market that offer static code analysis, namely:

- Coverity (https://www.coverity.com/)
- Code Climate (https://codeclimate.com/)
- Scrutinizer (https://scrutinizer-ci.com/)
- Codacy (https://www.codacy.com/)
- SonarQube (http://www.sonarqube.org/)

Only Codacy provides an option for dropping comments on GitHub. None of the existing solutions provide the option to automatically fix an issue or communicate via native GitHub comments, making user interaction very cumbersome for the users.

Income

The product will essentially be offered in two variants:

1. As a hosted service. GitMate runs on servers operated by us, providing a very easy set up experience for the users: everything already works with a simple login. Payment models could include fees per user or per repository. A free plan will be available for open source projects. This offers an easy approach and free trial for potential customers and has proven successful for similarly sized projects while supporting free and open source software.
2. As a software solution. Large enterprises usually operate their own GitHub Enterprise servers and want to continue using their internal infrastructure exclusively. In this case the software will be provided open source with a proprietary license.

Links

1. http://coala-analyzer.org/
2. https://help.github.com/articles/using-pull-requests/
3. https://developer.github.com/v3/
4. http://aws.amazon.com/
5. https://gitter.im/
6. https://docker.com
7. https://github.com
8. https://gitlab.com