



Intro

Bookie is an open source Python based bookmark application in the spirit of the old Delicious.com. The project has now grown into a whole set of libraries and tools that enable users to manage, find, and share their bookmarks.

We're looking for students interested in working on a project that involves backend and frontend code. Problems include issues around testing and continuous integration. Students will participate by writing unit tests for their work and going through code reviews. Your improvements will be available to hundreds of users of Bookie.

We're looking for **two** students to work on any of the below project or a project presented that we think will be valuable to Bookie and we can help mentor a student with.

Getting Involved

- Create an account on Bookie and see what it is and how it works.
- Grab the Chrome or Firefox extensions and use them to save some bookmarks.
- Check out the source code for Bookie and see if you can get it running.
- <https://github.com/bookieio/bookie> is the source code and has links to the docs.
- Check out the issue list and look at the bitesize bugs for small things to see if you can get into the code.
- Make sure to check out our HACKING doc for notes on a good Github workflow to work on Bookie. <https://github.com/bookieio/Bookie/blob/develop/HACKING.rst>
- Have fun and let us know if you've got any questions by sending an introduction email to the mailing list or asking around in #bookie on freenode!

Submitting a solid project application

When submitting your application for a project on Bookie make sure to keep a few things in mind.

Stand up meetings

Students will be expected to have several stand up meetings through the week with mentors on Google Hangouts. It's preferable if you have a camera and mic for this. The idea is to treat this

like a real developer shop and remote employees must keep in touch while working on the project to keep track of the state of things.

Note that mentors are in the eastern time zones and are willing to help find a decent time for all, but it might require some evening or morning meetings on the part of the accepted students.

Tracking tasks on Trello

Students will track their tasks using a Trello account. The mentor will help break the project into tasks with the student and all tasks will go through a pre-implementation stage, coding stage, code review, and finally a landing stage.

Let us know who you are

When you're submitting your application make sure to mention your name, email address, and irc nickname so that we can connect the dots and place who is applying with the great interactions we've had through the week.

Demonstrate some knowledge

When you're writing out your project proposal you should be able to specify areas of the codebase you know you'll need to understand. You should be able to have an intelligent discussion about what the project entails. You should also be ready to have a basic set of goals as well as stretch goals for the project, should time permit.

Provide some idea of what you hope to get out of the project

Make sure to share with us what you hope to learn and gain by performing this project. Are you hoping to gain some experience in writing tests? Better Javascript skills? What do you hope to take away so that the mentors can make sure you've thought about how the task is interesting and good for you and we can take that into account when building the task.

Links

code: <https://github.com/bookieio>

hosted version: <https://bmark.us>

docs: <http://docs.bmark.us/en/latest/>

mailing list: https://groups.google.com/forum/?hl=en#!forum/bookie_bookmarks

irc: #bookie on Freenode

twitter: [@mitechie](#) for the mentor and [@BookieBmarks](#) for Bookie

List of project ideas

Update Bookie to permit an Elasticsearch/Solr backend for full text indexing of content.

Mentor

Rick harding

Expected Outcome

Users can configure bookie to talk to an elasticsearch or Solr backend. The system will define a proper index definition for bookmark records including the web page content. The search interface and api will be updated to properly help users find their bookmarks via that fulltext content.

Skills

Elasticsearch/Solr/Lucene skills to determine how best to set it up for small scale use.
Python skills to write the search module and integrate it into the job system to get bookmarks into the index.
Javascript skills to update the search UX and update the Javascript API client.

Difficulty

medium

Update Bookie to do all readable processing of bookmarks with the bookie-parser web service.

Mentor

Rick harding

Expected Outcome

Setting up Bookie will actually run both the Bookie web application as well as the bookie-parser web service. The background tasks to find webpage content and index it will use the bookie-parser web service instead of doing requests on its own.

A stretch goal would be to extend the parser to do some additional functionality such as suggest tags to users based on the content.

Skills

Python to update the background work
Make/Scripting: to make bringing up the web service easy for new users.

Difficulty

easy

Create Juju charms for Bookie, bookie-parser, and create deployment bundles for them along with db backends (mysql or postgres) and setup a continuous deployment setup from github merged pull requests.

Mentor

Rick harding

Expected Outcome

A set of Juju charms with configuration options exposed would be available in the charm store. A bundle file would exist to allow users to deploy Bookie and all related services in one swoop. The charms would allow for scaling out from using sqlite for a database to mysql or postgresql.

Skill

Python and shell scripting to write the charms and tests.

Sysadmin skills for handling service config files and bringing up Linux servers running the services.

Jenkins skills to setup an automated test runner integrated with Github with automated landing of code reviewed pull requests that pass all tests.

Difficulty

medium

Create ability to monitor 3rd party sites for bookmark content. (Twitter, pocket, etc)

Mentor

Rick harding

Expected Outcome

Users can permit Bookie to monitor social networks or external services for urls they'd like to bookmark. Background tasks would monitor and add the bookmarks, fetch the content, and make sure they get into the system for the user automatically.

Skills

Oauth to allow Bookie to interact with 3rd party websites and services.

Python to update the background tasks that monitor those services and provide some UI for the users to add/remove/test the 3rd party services.

Javascript to assist with some of the UI requirements.

Difficulty

hard

Add private bookmark support

Mentor

Rick harding

Expected Outcome

Users can bookmark a url but mark that as a private bookmark that other users do not see on the website. Imports of bookmarks from other system would respect any indication that the bookmark is private.

Skills

Python to update the database code, apis, and views to handle the new idea of 'private'.

Javascript to update the UI of the site.

Difficulty

easy/medium

The task requires a lot of work to existing code, but the general idea of private-ness isn't difficult itself. The hardest thing will be getting very thorough test cases into the codebase to protect accidental leakage of private data.