Google Summer of Code 2018

https://summerofcode.withgoogle.com/

Application deadline: 17:00 GMT on Tuesday 23 January 2018
Deadline to contribute to this document: 14:00 GMT on Monday 22 January 2018

Bioschemas

Bioschemas aims to improve data interoperability in life sciences. It does this by encouraging people in the life sciences to use schema.org markup, so that their websites and services contain consistently structured information. This structured information then makes it easier to discover, collate and analyse distributed data.

Schema.org defines a range of properties and types, which describe a wide range of things that can be found on the Web – from concert tickets to automobile repair shops. The Bioschemas community is a global team that generates best practise guidelines for how to use the types and properties defined by schema.org to describe life sciences web pages and services. Where schema.org is unable to cover the more technical aspects of the life sciences, Bioschemas proposes widely deployed terms from the life sciences community that could be incorporated into schema.org.

The projects proposed within this document are designed to help our community extend its infrastructure and tooling thereby enabling improved data discovery through wider adoption of the markup. The projects can be grouped into tools to support data creation and tools to support data utilization.

Organisation Administrators

- Alasdair Gray

- Rafael Jimenez

Mentors

- Dominique Batista
- Niall Beard
- Manuel Bernal Llinares
- Justin Clark-Casey
- Alexander Garcia
- Leyla J Garcia
- Olga Giraldo
- Alasdair Gray
- Federico Lopez
- Kenneth McLeod
- Sarala Wimalaratne

Project Ideas

Please list and define below your ideas. At the end of the document you can find a template with the sections to be completed for each idea. For more information and advice on how to define a project idea check the GSoC guide.

Bioschemas Common Crawl

Crawl websites to create a Bioschemas markup dataset that can be used by other projects.

Description

Various projects (such as Buzzbang and SPARQL over Bioschemas) want to make data across the whole of life sciences easier to find and use. To do this using Bioschemas, they will need a crawl of all the webpages that host Bioschemas markup. Rather than each project producing this crawl separately, it will be more efficient if they can share a common dataset, which may be supplemented with other datasets such as Common Crawl.

Expected outcomes

- A crawler for Bioschemas data, likely based upon an existing web crawler such as Apache Nutch.
- A downloadable RDF dataset containing the crawl.
- A periodically refreshed crawl.

Skills

- Java
- Devops
- RDF
- Bonus: any experience with Triplestores

Difficulty

- Hard

Mentors

- Justin Clark-Casey

Buzzbang

An open-source Google-like Bioschemas-enabled search engine.

Description

Even though life sciences database projects have produced many sophisticated query interfaces, simple free text search, in the style of Google's search frontend, is still extremely popular for finding scientific data. The Buzzbang project looks to leverage Bioschemas markup to provide this kind of simple search interface to return more relevant and interconnected results than is possible with just the analysis of free text. This project encompasses work to improve both the Buzzbang web frontend and the backend operations to create the search index.

Expected outcomes

- Improved backend indexing for Buzzbang, possibly replacing some existing custom crawl scripts with processing of a common crawl produced by the "Bioschemas Common Crawl" project.
- Improved frontend for the Buzzbang project, with display of Google-style rich results generated from Bioschemas markup.

Skills

- Java
- Python
- Solr
- Flask

Mentors

- Justin Clark-Casey

Difficulty

- Hard

References

- http://buzzbang.science/
- https://github.com/justinccdev/bsbang-crawler/tree/dev

- https://github.com/justinccdev/bsbang-frontend

SPARQL over Bioschemas

Implement a SPARQL interface over crawled Bioschemas data

Description

The Linked Open Data cloud is a vast distributed graph of interconnected structured information, a large proportion of which comes from the life sciences. This information is expressed in a common data structure called the Resource Description Framework (RDF), which enables distributed querying using the SPARQL standard. Bioschemas markup itself can be expressed as RDF. This project would create a SPARQL endpoint that spans crawled Bioschemas data from the "Bioschemas Common Crawl" project, to join it to the Linked Open Data cloud and make this data easier to combine with other information.

Skills

- RDF
- OWL
- SPARQL
- Triplestore databases

Mentors

- Federico López
- Leyla Garcia
- Alexander García

Difficulty

- Hard

References

- https://github.com/BioSchemas/bioschemas-nutch-indexer

Map2Model

Automating the generation and publication of Bioschema's specifications from best practise guidelines.

Description

This tool supports our community in the creation and display of the specification documents that define a profile. To support widespread collaboration, we use Google Sheets for the definition of a profile. The Map2Model tool merges the contents of the Google Sheet with pre-existing types from Schema.org, before converting the result into markdown which can

then be rendered on the community website for use by other Bioschemas tools and the wider life science world.

Expected outcomes

- Convert Map2Model from a standalone process to a web service that is run in response to a request from the community.
- Make the Map2Model process more robust.
- Automatically publish the output as a versioned specification on the Bioschema's web page.
- Generate machine processable constraints defining the profile.

Skills

- Web services
- Devops
- Java or Python
- RDF (desirable but not essential)

Mentors

- Alasdair Gray (http://www.macs.hw.ac.uk/~ajg33/)
- Kenneth M^{c}Leod (http://www.macs.hw.ac.uk/~kcm)

Rate difficulty

- Medium

References

- https://github.com/BioSchemas/map2model
- http://bioschemas.org/specifications/

Markup Builder

A web application for prototyping markup against the Bioschemas profiles.

Description

This web application supports users in the creation of Bioschemas compliant markup required for inclusion on their web resource.

- Bioschemas provides profiles for schema.org mark-up in order to structure and expose life-sciences metadata on the web. Each profile brings a list of allowed attributes with their constraints and properties. Some attributes are required, some are composite, some allow multiple values, some are under controlled vocabularies

and some can even be all of that. We want to create a web application that assist users into the creation of their metadata structure, through dynamically generated forms, allowing an easier sharing over the web.

Expected outcomes

- Needs:
    - Help us get the profiles attributes properties loaded into the app from different file formats.
    - Help us build responsive widgets that can be re-used for different attributes and profiles.
    - Link these widgets to the relevant attributes in the JSON-LD output.
- Project results:
    - Specifications files can be loaded into the app to dynamically generate forms.
    - The generated widgets allow to add new attributes to a JSON-LD variable based on loaded properties. The JSON-LD variable is displayed in real-time (without form validation).
    - The code should be importable as an AngularJS library in order to be reusable

Skills

- Languages: JS / CSS / HTML
- Formats: JSON-LD
- Framework: AngularJS.

Mentors

- Dominique Batista
- Alasdair Gray

Rate difficulty

- Medium+

References

- [This is optional, add if you have any relevant references]

Validata

A web application for validating Bioschema's markup against the specifications.

Description

This web app supports users in testing whether their generated markup is compliant with the Bioschemas specifications. The validation engine currently supports machine processable constraint descriptions written in the Shape Expression (ShEx) language which capture the

Bioschema profiles. The Validata app allows users to supply some markup which is then compared to the profiles and generates an error report to the user for the snippet of code they provide.

Expected outcomes

- Extend the existing system such that:
  - it supports JSON-LD;
  - it can validate an entire site; and,
  - it is able to retrieve pages from an external URL so that they can be tested *in situ*.
- Improve the user interactions with the application making it easier to use and simpler to act upon the validation reports.

Skills

- Languages: JS/CSS/HTML
- Formats: JSON-LD

Mentors

- Alasdair Gray (http://www.macs.hw.ac.uk/~ajg33/)
- Leyla Garcia
- Kenneth McLeod (http://www.macs.hw.ac.uk/~kcm/)

Rate difficulty

- Medium+

References

- https://github.com/HW-SWeL/Validata

Bioschemas4JATS

How could we have infoboxes describing the content of a scientific publication?

Description

The Journal Article Tag Suite (JATS) is an XML format used to describe scientific literature published online. We want to define the representation of JATS data elements in schema.org and have this as an addition to BioSchemas. This projects requires an engaged critical thinker; the task at hand needs more analysis than coding. The student will analyze the JATS specification, determine what data elements from that specification describe the publication and the content and then, map these to Bioschemas. If needed, then the student will propose new data elements. If possible, the student will also design and implement the

corresponding parser, making it easier to extract from JATS those data elements that are part of the resulting Bioschemas specification.

Expected outcomes

- Mapping JATS data elements to Bioschemas
- Defining additional elements in Bioschemas that make it possible to describe a scientific publication
- Bioschemas specification for the JATS format.
- Parser, from JATS to Bioschemas4JATS

Skills

- Curious and willing to learn
- Basic knowledge of JSON-LD, XML
- Basic to medium Python; how to write parsers, basic data wrangling.
- Familiar with github

Mentors

- Alexander Garcia (https://github.com/alexgarciac, https://www.researchgate.net/profile/Alexander_Garcia)
- Olga Giraldo
- Leyla Jael Garcia

Rate difficulty

- Easy+

References

- https://jats.nlm.nih.gov/

Template for project ideas

Copy the template below if you want to describe a new project idea. More information about the content of a project idea can be found in the GSoC Mentor guide.

[Project title]

[Optional subtitle: one short sentence description]

Description

[2-5 sentences]

Expected outcomes

- …

Skills

- ...

Mentors

- ...

Rate difficulty

- [Easy, Medium or Hard]

References

- [This is optional, add if you have any relevant references]