2022-05-24 - VIVO Development IG

Date

24 May 2022

Call-in Information

Time: 10:00 am, Eastern Time (New York, GMT-04:00)

To join the online meeting:

- [https://lyrasis.zoom.us/j/84378615572?pwd=84378615572#,,,,*556561# US (San Jose)](https://lyrasis.zoom.us/j/84378615572?pwd=84378615572#,,,,*556561# US (San Jose))
- [843 7861 5572](https://lyrasis.zoom.us/j/84378615572?pwd=84378615572#,,,,*556561# US (New York))

Dial by your location

- +1 669 900 6833 US (San Jose)
- +1 929 205 6099 US (New York)
- +1 253 215 8782 US (Tacoma)
- +1 301 715 8592 US (Washington DC)
- +1 312 626 6799 US (Chicago)
- +1 346 248 7799 US (Houston)
- 877 853 5257 US Toll-free
- 888 475 4499 US Toll-free

Meeting ID: 843 7861 5572
Passcode: 556561

Find your local number: [https://lyrasis.zoom.us/u/kerqtGDrJ4](https://lyrasis.zoom.us/u/kerqtGDrJ4)

Slack

- [https://vivo-project.slack.com](https://vivo-project.slack.com)

Attendees

- Dragan Ivanovic
- Brian Lowe
- Georgy Litvinov
- Michel Héon
- Benjamin Gross
- Benjamin Kampe
- Veljko Maksimovic
- José Ortiz
- Andreas Czerniak
- Mark Vanin
- Nicolas Dickner
- William Welling
- Huda Khan

Agenda

1. Questions/Issues/Pull requests/Announcements
   b. VIVO 1.13.0 release candidate - [Release Testing - 1.13.0](https://vivo-project.slack.com/archives/C8RL9L98A/p1653394256098719)

2. The next sprint
   a. date
      i. May 30th - June 17th
   b. Registration
      i. link
   c. Infrastructure
      i. Wiki page - [Sprint - Dynamic API based on an ontology](https://vivo-project.slack.com/archives/C8RL9L98A/p1653394256098719)
ii. Slack channel created
iii. Project board
   1. We will continue using this Project board - https://github.com/orgs/vivo-project/projects/2
      a. but extend it with new issues (features, bugs)
iv. Starting point
   1. backend
      i. will be renamed to dynamic-api
      i. wouldn't be used
   b. frontend
v. Cleaning up not completed work at previous sprint
      a. GET all and GET one - two dynamic actions or one ?
   2. https://github.com/vivo-project/Vitro/pull/286
      a. deletion and modification
   4. Transforming xml notation of Dynamic API ontology to turtle/n3 notation
   5. Introducing new jena model for dynamic actions
   6. Cleaning up issues at https://github.com/orgs/vivo-project/projects/2/views/1
vi. Tracks
   1. Ontology
      a. RDF representation of dynamic actions and resourceAPIs
   2. Binding
      a. Binding to Java objects representation
   3. Execution
      a. Execution of a dynamic action
   4. RCP Endpoint
   5. REST Endpoint
   6. Security
      a. Definition of dynamic action roles
      b. Authentication (endpoint and UI)
      c. Authorization
         i. JWT
      ii. Do we need OAuth2 or OpenID Connect ???
      d. The ontology - link
    7. Validation
      a. syntax,
      b. over the ontology,
      c. generic SHACL rules for all dynamic actions
      d. custom SHACL rules for specific dynamic actions ???
   8. UI for running a dynamic action
      a. Angular & Angular Material
      b. Some notes regarding angular from William Welling.
         i. I would recommend against using a openapi specification generator
            1. it likely does not take into consideration angular universal for SEO, central state
               management, or i18n
            2. it is not likely to be a long lived dependency, popular, or actually work well with the
               dynamism of custom actions
         ii. choose between the server side module (both express and angular universal)
            2. https://github.com/nestjs/ng-universal
         iii. consider a central state management up front
            1. https://ngxs.io/
      iv. choose i18n library (important to know whether we want runtime or build time i18n)
         2. https://ngneat.github.io/transloco/
         3. https://angular.io/guide/i18n-overview
   v. adopt a style guide and have it enforced using linting
      1. have github action check code styles
      2. protect main branch against style guide violations
   vi. incorporate best practices and have test coverage in initial commit
      1. have github action that performs code coverage
      2. protect main branch for code coverage decrease
   9. CRUD operations for custom actions
   10. UI for management of custom actions
   11. Exploitation

Notes

Benjamin Kampe said the German VIVO workshop will be organized as an online event open (free) for participation, but majority of presentation might be in German.

Dragan Ivanovic informed participants that VIVO 1.13.0 release candidate has been created, waiting confirmation from some committer that release candidate can be deployed and run for spreading the news.
**Task List**

**Previous Tasks**

- **Dragan Ivanovic** to open an issue for a reverse proxy (redirecting) - https://vivo-project.slack.com/archives/C8RL9L98A/p1651739159126489
- **Dragan Ivanovic** to contact Andrew Wood for resolving the issue with registration for the Slack channel
- **All** to consider participation in the sprint and if decide to participate to register for the sprint - [link](https://github.com/vivo-project/VIVO/issues/3698)
- **Dragan Ivanovic** to send a require to add in the Leadership group agenda a topic for discussing Copyright owner in License
- **All please help in filling this document** - [spreadsheet](https://drive.google.com/file/d/1N9789y7P2fWbIoGmIwX4YyUJ9Q4ZI5e2/view?usp=sharing)
- **Dragan Ivanovic** to create a draft version of a Google Spreadsheet document with labeled columns representing requirements and to share with committers
- **Committers** to review the structure of the document and to fill it with data for JS and CSS frameworks

---

**Georgy Litvinov** noticed a problem of duplication of some nodes in localization files and loading of those nodes. It should be fixed before VIVO 1.13.0 is published. **Brian Lowe** will try to create a PR which fixes this issue, **Dragan Ivanovic** will organize reviewing of the PR, and create a new VIVO 1.13.0 release candidate.

**Dragan Ivanovic** proposed an ontology for definition of permissions (https://webprotege.stanford.edu/#projects/88e4b33-30b4-4486-8bd3-fa9dd60cf9f3/sharing) which should be in accordance with discussion at pre-sprint meeting and should address granting permissions for execution and modification of a dynamic action. **Georgy Litvinov** thinks this is a big issue deserving a complete sprint, and we need a quick and simple solution for forthcoming sprint. That solution might be that users with role Administrator can modify (create, update, delete) dynamic actions, while a role for execution of a dynamic action is defined by using hasRole object property in the Action class.

**William Welling** explained why angular code generator is not solution for dynamic api, and why he thinks we need dynamic routing and dynamic UI capable to adopt at any change in a dynamic action, or adding a new dynamic action. This will be a challenging task which need further exploration whether and how it is possible.

Draft notes in Google Docs