

Posudek diplomové práce

Matematicko-fyzikální fakulta Univerzity Karlovy

Autor práce Altynbek Orumbayev

Název práce Decentralized Web-based Data Storage for LinkedPipes Applications
using Solid

Rok odevzdání 2020

Studijní program Informatika **Studijní obor** Softwarové a datové inženýrství

Autor posudku Petr Škoda **Role** oponent

Pracoviště Katedra softwarového inženýrství

Text posudku:

LinkedPipes Applications (LPA) is a successfully defended software project created at MFF UK. This thesis aims to implement Solid-based data Storage for LPA.

The topic can be seen as a more difficult one, not because of the amount of code or complexity of the problem, but because of the Solid ecosystem. The technology and specification of Solid are still work in progress; therefore, they are harder to use compared to other more mature projects. In addition, the student also needs to understand concepts such as ontology modeling, RDF and OWL.

The implementation has three main parts: OWL vocabulary, LinkedPipes Storage (LPS) library, and React components for LPA. LPS effectively wraps Solid pod so it can be easily used from LPA. The React components are connected to LPA. This makes the LPS the main standalone code output of the thesis. The code quality, tests, and used tool-set are up to date and of reasonable quality. Many of these qualities seem to be consequences of the LPA software project. This can be showcased on non-functional requirements that are mostly inspired by LPA. On the contrary, the GitHub repositories hosting the code contain few todo-like texts and commented legacy code, perhaps showing a lack of time or attention to detail.

In contrast to overall well done code part of the thesis the text part is not up to the same standard. This is even more noticeable as the difficulty was not in the code part and as the text part should define the contribution and distinguish the thesis from the LPA software project.

The first issue is the connection to LPA, where the text alters between describing principles and requirements of LPA and LPS. This probably comes naturally as the student is also one of the LPA authors, but I think there should be a better distinction between roles of LPA developer and author of the thesis.

The high-level flow of the text is good and includes all that is necessary i.e. related work, use-

case, architecture, implementation, user documentation, and developer documentation. However, the content of the chapters is not of the same quality. This can be demonstrated on following examples:

- The mini-survey of Solid alternatives is more of a general information rather than an evaluation of requirements from the perspective of the thesis.
- Non-functional requirements are not well-defined and they are heavily inspired by LPA.
- The implementation chapter is a continuation of the architecture chapter.
- Parts of what may be a user documentation is in the implementation section.
- There is a section with UI description of used tools.

The text is not easy to read or follow. The reasons for that are grammar mistakes, logical gaps/jumps in the text and out-of-context diagrams/figures. It thus remains only a minor note, that the diagrams and figures are not using vector graphics.

Overall the topic is ambitious and complex, the code is of a reasonable quality, but the text part does fail to deliver an expected quality on several levels.

Práci doporučuji k obhajobě.

Práci nenavrhuji na zvláštní ocenění.

V Praze dne 27.01.2020

Podpis: