HWDB Python Upload Tool Updates

Alex Wagner

March 14, 2023

UNIVERSITY OF MINNESOTA
Driven to Discover®

REST API

- Hajime and I have been testing the new v1 of the REST API and have been sending feedback to Steve and Vladimir
- We love the new Swagger documentation that allows us to test things directly from the documentation web pages!
- We have been collecting our thoughts/questions/comments in a shared Google doc
    - https://docs.google.com/document/d/1Ljcf20-nSoEDwon_cJS_SSGK68MDmgIGUbxV-1nbBSc/edit?usp=sharing
- Vladimir has already made some of the changes for us. Thanks, Vladimir!!

Recent Changes

- Code for the docket encoder is now reorganized and broken into smaller pieces that can be individually tested without needing to go to the database
- Unit/regression tests have been added for the docket encoder
- A new library for interfacing with v1 of the REST API has been created
    - Some of the new methods are compound methods that make multiple REST API calls and combines them into a single output, e.g., getting a system/subsystem/component type tree
- Some unit/regression tests have been added for the v1 interface

Future changes

- Compound methods in the new v1 interface will need to be updated to make REST API calls concurrently instead of sequentially
  - This isn't exactly trivial, but I already have some code I wrote back in October/November that does this for a specific case, so I just need to work out how to generalize it
- Transparent caching of data, to avoid unnecessary calls to the REST API
- More unit/regression tests are needed to cover all the various options available in the docket encoder format
- Thorough documentation needs to be written, with many examples provided
- There is a lot of code currently in the project that has been started on but is not currently used by the tool, and a lot of code that is outright obsolete, which really ought to be cleaned up before releasing the code to others
- I've somewhat butchered the GitHub project I was storing this code in, so I either need to fix it or start a new project

Possible future considerations

In the very near future, most of my time will be working on the iPad app with Hajime, but at some point in the future, there are some features that I think would be extremely useful:

- Database integrity tool – given a part type id, examine existing components for validity and completeness, e.g., if the part is supposed to have parents or children, have they been linked? If there are tests defined for a part type id, do the components for this type all have tests uploaded?
- GUI for assisting with creating encoders – how this would work is still a bit vague in my mind, but the idea would be to select some CSV files that contain a sample of the data you would like to upload, and *[some magic happens]* and it creates the encoder for you. Also, it would be able to fetch/retrieve these from the spec defs in the database
- GUI equivalent of the docket upload tool
- Simplified GUI tool providing some similar functionality as Hajime's iPad app, e.g., the ability to add components directly, without needing to supply a CSV

Thank you for your time!

Questions? Comments?

UNIVERSITY OF MINNESOTA

Driven to Discover®

Crookston Duluth Morris Rochester Twin Cities