# Domain Analysis Document

**DAD Team**

**Jul 30, 2020**

# CONTENTS:

# DAD OUTLINE

- Stakeholders
- Use cases
    - Driver use cases
    - Community use cases
- Phenotypic data & information
- Entities
    - Core entities
        * Entity X
        * Entity Y
    - Other entities
- Phenopackets
- FHIR
    - Phenotypic data representation in FHIR base specification
    - FHIR mapping
- Tools and resources
- Glossary

*Some term* is defined in the glossary.

## 1.1 Contributing to the DAD

Contributing to the DAD will generally be through GitHub pull requests. Pull requests must start from the master branch, not another branch.

Perform the following initial steps to setup your environment setup:

1. A compatible Python version is required. We recommend you use a tool that simplifies managing your Python version. This project is setup to use pyenv https://github.com/pyenv/pyenv

2. It is also recommended to use Python virtual environments. See: https://docs.python.org/3/tutorial/venv.html

3. Git clone (after a fork on GitHub if needed) https://github.com/phenopackets/domain-analysis

The above are one time steps. Subsequent to that, perform the following steps with each new contribution:

1. Switch to the master branch with `git checkout master` if needed.

2. Perform a `git pull` to get the latest changes.

3. Start a new branch for your changes with `git checkout -b your-new-branch-name`.

The above will get you setup on a new Git branch for your new contribution. If you already have a branch that you need to continue to work on, you only need to checkout that branch with `git checkout your-branch-name` and continue editing. However, either way, you still have not activated the development environment yet. To do that you need to follow the steps below. We recommend you do that every time you switch or create a branch to make sure your environment is up to date.

1. Assuming you are using Python virtual environments, and on a Linux like machine, run `. venv/bin/activate` from the repo's root directory to enter the virtual environment.

2. Run `pip install --upgrade -r docs/requirements.txt` in the root of the Git repository.

3. If you'd like to have live preview of your edits (when they are saved), run `sphinx-reload docs/` in the root directory. This should perform an initial build, open the home page in your browser, and watch for any saved changes. Any changes will cause your browser to reload the page with the new edits.

4. If you'd like to perform occasionally builds manually when needed and open the relevant built files from `docs/_build`, you'd run `make html` from the `docs/` directory.

Once your edits are ready for commit and push, follow the usual Git and GitHub steps to push your branch and start a pull request. All pull requests will be built by Read the Docs and you will be able to see a published version of your changes before they area accepted into the master branch.

## 1.2 Glossary

**Some term** Some term's description.

# INDEX

## S