
plone.importexport

Proposal for Plone Foundation as a Sub-Org under PSF

By: Aalekh Jain

Email: -

[GITHUB](#), [RESUME](#)

University : [International Institute of Information Technology, Hyderabad, INDIA](#)

Programme : Pursuing B.Tech in ECE and MS by Research in Computer Vision and Machine learning

Contact number: -

Project Context ([LINK to Project Discussion](#))

Currently webmasters, the primary users of this addon, find it difficult to import/export data from Plone sites and usually end up with raising requests in community support for the same. So an addon which would provide an easy to use import/export solution with ease UX is scoped here.

Aim

The aim of the project is to develop a method, easy-to-use by nontechnical and technical user to import content, export content or move content securely between sites (either Plone or other CMS) using an interactive online UI. A fully polished release as an installable add-on and possibly accepted into Plone 6 as a core component is what mark the success of this project.

Scope and Major Challenges of this Project

The underlying idea of this project is to develop an addon which would provide an import/export solution even for non-techy users with ease to use UX.

The following scopes are not yet finalised and need more community discussion to make it real. Therefore there may be a few changes in the original programme.

- ❖ Providing an easy to use interface which is built on top of the Plone restapi.
- ❖ Handling permissions on the imported/exported content. (Maybe Plone restapi can handle permission as mentioned in their docs, I'm not fully sure here)
- ❖ Developing a mechanism to handle existing content and perform required action on them.
- ❖ Handling large files during import/export.
- ❖ Providing a solution for bulk update of required fields/contents.
- ❖ Integrating useful error logs (if any) along with the import/export tasks.

-
- ❖ Support to all the Plone 5 default content-types and any regular Dexterity type.
 - ❖ A fully polished release as an installable add-on and possibly accepted into Plone 6 as a core component.
 - ❖ Writing Docs to explain working of features and to get an overview of the addon

Why did I choose this project and why should I be selected for this? Have I spoken to anyone in the community about the project?

To be honest, I am relatively new to the community. However I have been learning about Plone over the past weeks and played with it on my local machine. I have gone through most sections of Plone [docs](#) and [training](#) and spent almost a week with them and which I feel to be the first point of contact for any newbie developer to Plone as they really get the programming sense of Plone's development into their head.

I have an elementary understanding of plone.importexport and plone.restapi addons which will play an important part in this project.

Also, I'm participating in ongoing [discussion](#) of this project.

Do I have enough time for this project? Do I have any other commitments during the period ?

Yes, I have ample time during summer for this project for the time period from May to July, which is my college's summer holidays.

Most interesting programming projects I've worked on-

- ❑ I'm the second lead contributor (after jdalton, creator of lodash) of a JavaScript library, *is.js* having around 9k stars on github, written in JavaScript. ([link](#))
- ❑ I was the portal admin of our college's annual techno cultural festival *Felicity* (in year 2016-17). The portal is used for hosting all the online programming events (the events cater to many of the fields of computer science, some of which are: competitive programming, capture the flag, parallel programming, machine learning etc.) which are organized as part of Felicity.
- ❑ I've written a tool in python and bash that allows to write any desired name on your github profile ([link to github repo](#))
- ❑ Apart from these, I've developed various scripts for automating tasks in python and javascript. Some of them are: Controlling electrical devices in a room using a web interface (written in arduino and python), automating the process of sending like invites for promoting a facebook page and various other social media websites, I've written addon in firefox for the same purpose as well, scraping data from several websites using frameworks like jsdom, selenium, mechanize etc.
- ❑ I've also worked as Algorithmic and scripting problem setter (which were majorly based on data structure, algorithms and python and bash scripting). I'm active on various competitive coding websites like [codeforces](#), [codechef](#), [leetcode](#).

Most of the above projects and activities were done for learning purpose and for fun. Some of the projects and responsibilities that were related to my coursework are as follows -

- ❑ A highly parallelized generative, unsupervised clustering and voting based object/human tracking algorithm. This was done as part of my research project in the year 2017.
- ❑ Teaching assistant for the course Software Engineering (2017) and Computer Programming (2018)
- ❑ Building a real time ground control system for IIT-H team PegaSAT for American Astronautical Society organized, NASA affiliated competition, *CANSAT*, in year 2015. Our team stood position in top 15
- ❑ Apart from this, I'm currently working actively on the projects related to Computer vision and machine learning as part of my research work.

I'm a huge supporter of open source and always love to contribute and promote open source whenever I get opportunity. In my college, I'm a promoter and a member of open source development group ([OSDG](#))

My Development methodology

I am more comfortable working in an agile manner. Therefore once a idea/method gets solidified with consent of my mentors I would code it along with testing. Also at each point, I would update my mentors. in case of any confusion I would take it up to discuss with community or follow past thread with similar issues.

Skills and tools required

Language -python, javascript and basic programming skills

Tools - Git, CI, Selenium and Pycharm for python development environment

Pre-work for this project (before coding sprints get started)

To get clear understanding of the task, I would first like to explore Plone schema, restapi and datatypes (dexterity and archetypes), as they are the building blocks of this project, responsible for both export and import. Later I would explore the existing addon (plone.importexport developed during GSOC'17) to list out enhancement and fixes required and freeze a methodology to do the same.

Timeline

Starting from 6th May'19 and ending on 26th August 2019. Work order for each feature:

- ❑ writing implementation and improving upon existing work
- ❑ writing tests (functional, acceptance and integration)
- ❑ writing documentation (how the feature should work)
- ❑ updating docs and tests

Deliverables and challenges(risks) for each time period are below with each sprint having 10 dedicated days of work following the work order as mentioned above. I have also included enough buffer time to overcome the challenges mentioned :-

- ❖ Community/Mentor Bonding Period, May 6 - May 27
 - Deliverables
 - Sprint 1. Getting comfortable with existing add-on as mentioned in pre-work to understand the underlying mechanism
 - Sprint 2 - Solidify scope and methodology to be used in this project. This is more towards community discussion.
- ❖ 1st Coding period, May 27 - June 28
 - Deliverables:
 - Sprint 1 - Existing code cleanup, setup CI, write tests for existing features
 - Sprint 2 - build initial documentation, preparing mockups, fix failing tests, generate error logs
 - Sprint 3 - Buffer time and implementing suggested changes
 - Challenges:
 - A run through existing code and fix failing cases
 - Learn building documentation and testing in plone environment
 - Exploring security measures to handle permissions
 - Exploring the support for Plone versions < 5
- ❖ 2nd coding period, June 29 - July 26
 - Deliverables:
 - Sprint 1 - Develop UX as planned from previous sprints
 - Sprint 2 - Developing a mechanism for dry run
 - Sprint 3 - Buffer time and implementing suggested changes
 - Challenges:
 - Ease for user interaction
 - Exploring Dry run possibilities
- ❖ 3rd coding period, July 27 - August 25
 - Deliverables:
 - Sprint 1 - Give a support to bulk update and large import/export files
 - Sprint 2 - First release of add-on along with code and docs enhancement
 - Sprint 3 - Buffer time for any pending tasks, code and docs enhancement
 - Challenges:
 - Efficient methodology to support large files
 - Efficient methodology to allow bulk updates
 - Addon release
 - 100% PEP8 coverage
 - 100% test coverage
 - 100% documentation coverage

❖ Final Evaluation, August 26 - September 2