

Google Summer of Code Proposal

Hi,

I am Arun H, I am interested in contributing and improving the GNU Mailman software application.

Contact:

Email: arunhari3210@gmail.com

IRC Nickname: arun3

Country: India

Time Zone: GMT +5:30

Github: <https://github.com/Arunhari333>

Gitlab: <https://gitlab.com/arunhari>

About Me

I am a 2nd year student at Model Engineering College, Kerala, India, pursuing a Bachelor's degree in Computer Science. I am passionate about coding. I started learning html, css and javascript from freecodecamp.org . I learned python and it is currently my favourite programming language. I started understanding website back-end working by creating a micro-project using Django. I am a member of the FOSS club in my college and love open source software.

Project Information

Project Title:

Support for REST Callbacks in Core

Brief Project Summary:

This project has many subprojects like Mailman Core, Mailman Client, Django-mailman3, Postorius, HyperKitty and other subprojects, each aimed for specific functionalities of the Mailman application. This project aims in conveying the

changes in Mailman Core to the clients in an efficient way. It aims in increasing intersubproject communication by implementing a pub-sub model on the Core to which other subprojects like Postorius and HyperKitty could subscribe to as clients.

Detailed Project Description:

There are various ways to implement the publisher-subscriber model in the mailman core. The ones most suitable in my opinion are:

Using Redis

Redis is popular and easy to use. The model would look like so:

Publisher:

```
import redis
import sys

if len(sys.argv) == 3:
    program, environment, action = sys.argv
    client = redis.Redis(host = 'x.x.x.x', port = y)
    client.publish(environment, action)
else: .....
```

Subscriber:

```
import redis

env = '...'

client = redis.Redis(host = 'x.x.x.x', port = y)
p = client.pubsub()
p.subscribe(env)

while True:
    message = p.get_message()
    if message and not message['data'] == 1:
        message = message['data'].decode('utf-8')
        print(message)
```

Using Microservices:

Another method is using microservices(yosun and RabbitMQ) , I found this to be easier to implement with django.

Publisher:

```
from kombu import Connection, Exchange
from yosun import Yosun

# Define Connection
connection = Connection('.../')

# RabbitMQ Exchange definition
exchange = Exchange('events', type='topic')

# Yosun initialize
yosun = Yosun(connection, exchange)

yosun.publish('a.x', {'name' : 'abc'})
```

Subscriber:

```
from kombu import Connection, Exchange
from yosun import Yosun

# Define Connection
connection = Connection('...')

# RabbitMQ Exchange definition
exchange = Exchange('events', type='topic')

# Yosun init
yosun = Yosun(connection, exchange)

def on_x(body, message):
    print(body)

def on_a(body, message):
    print(body)

sub = yosun.subscribe('a.#')

sub.all(on_a)
```

There are other options such as Django-channels. It can also be done without these softwares, but I believe that wastes time. Any way is fine with me.

Timeline / Plan of action

If selected, I shall be very regularly active on the Mailman communication channels and be open to communication and mentor feedback either through Mailman or through IRC. Here is a breakdown of how I plan to use the time available to me:

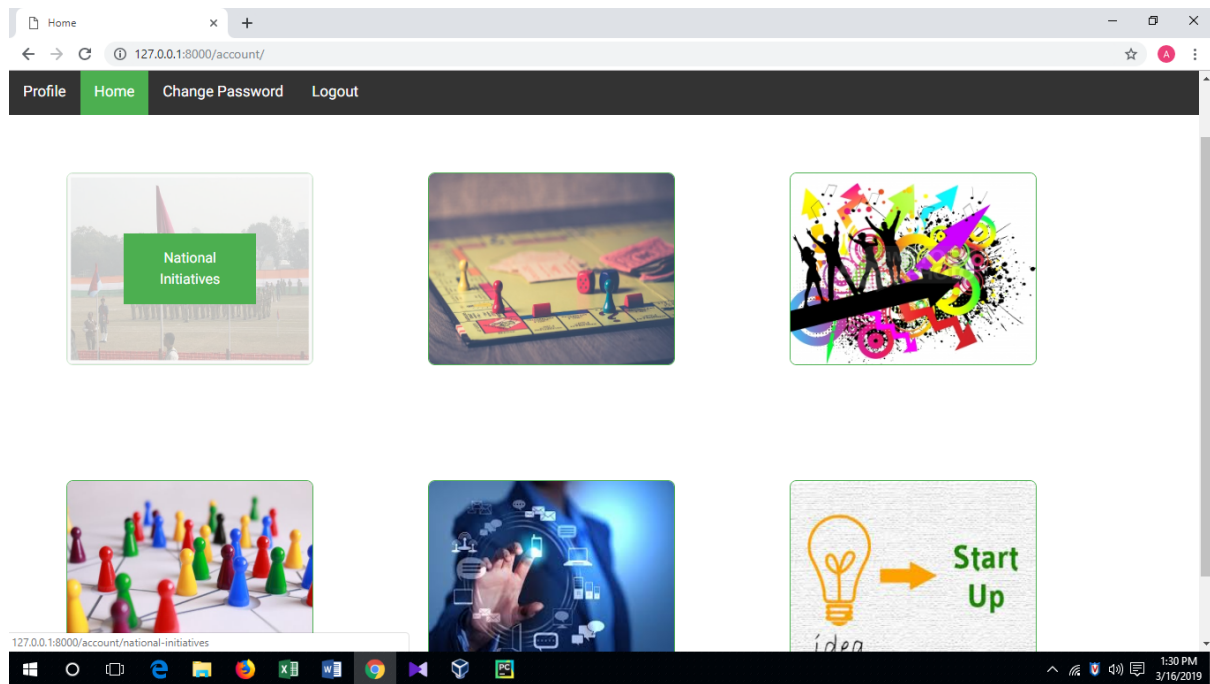
- **Now to May 6th:** Familiarize myself with the community and the various mailman projects.
- **May 6th to May 27th:** Form clear ideas and visualizations of the task at hand, share and discuss with my mentors and the community to form final decisions.
- **May 27th to June 12th:** Complete all tasks that does not require major work.
- **June 12th to July 2nd:** 60% of the major tasks are completed.
- **July 2nd to July 20th:** Finish all the major and minor tasks.
- **July 20th to August 4th:** Polish, optimize and check everything that was added.
- **August 4th to end of GSoC submission time:** Finish any outstanding tasks.

Time Availability

I have university exams on June, during which I will be able to put in only light work, but after that I have no commitments and will be able to spend 45+ hours a week for this project until the end of GSoC.

Why Me?

I am very comfortable coding in python and django as I have done a micro-project using django. The aim was to create an application to store the activity points of students when they select their activity and upload the certificate that proves it.



The link to the micro-project: <https://github.com/Arunhari333/Django-creditZup>

I learned about api and rest framework afterwards as this project does not require it.

Over the past weeks, I familiarized myself with the different projects of Mailman and solved 2 issues and is currently working on another.

(https://gitlab.com/mailman/postorius/merge_requests/394)

I am looking forward to contributing to this exciting project and will continue being a contributor beyond GSoC. Thank you for reading my proposal, I hope that my summer would be spent doing something that matters.