Sunbird Serve

Installing Serve Beta Platform

📄	Prerequisites

📄	Serve Beta on Windows

📄	Serve Beta on Ubuntu

Prerequisites

This is an installation guide for serve beta version.

Versions: Python Version - 2.7.18

   pip version - 20.3.4

Don't copy-paste the `$` signs, they indicate that what follows is a terminal command

Serve Beta on Windows

Windows:

- Download and Install Visual Studio Code or any code editor from here
- Clone the git repo directly to VS code editor.
  Open VS code editor window, select the source control from the left hand side corner.
  Select "Clone Repository" and provide the git repo url mentioned here.
- Download and Install python 2.7 version here.

==

Creating and Setting up of Virtual environment

Create virtual environment to install all the necessary applications and dependent libraries within the virtual environment.

Open the terminal in VS Code

```
$ pip install virtualenv
$ virtualenv env --python=python2.7
$ cd env
$ \Scripts\activate

//The virtual environment is now activated.
```

==

Install pip and Django

Check pip version: python -m pip --version

If the pip version is not 20.3.4 then upgrade the version to 20.3.4

If pip is not installed, install pip by the below command.

```
$ apt install python-pip 20.3.4
```

Install Django:

```
$ pip install django==1.4 django-mailer==0.1.0 pip south
```

Setup PEAK rules – required for some of the generic functions used in python.

```
$ cd PEAK-Rules-0.5a1.dev-r2686
$ python setup.py install
$ cd..
//navigate back to previous folder
```

Install Remaining Packages

```
$ pip install turbogears xlwt registration notification django-notification==0.2
django-social-auth==0.7.28 Pillow reportlab xhtml2pdf==0.2.4 html5lib==1.0b8 xlwt pisa

$ pip install django-registration==0.8 xlrd python-social-auth==0.2.14
$ pip install pydrive (If version error, then install rsa==4.0) (global)
$ pip install feedparser chronograph simplejson python-dateutil

$ pip install reportlab pypdf2 google-auth-oauthlib boto3~=1.16.57 python-dotenv
$ pip install django-cors-headers==0.11 pyfcm openpyxl redis
```

==

Database Setup

Download mysql package and install here.

start mysql server and connect to localhost instance - Mysql@localhost:3306

Once connected, within the Administration tab select "Data Import/Restore" .

Select Import from the self contained file and provide the path for the .sql file.

Within the default target schema, create a new schema and select the new schema as the target db.

Select 'Start Import'

.sql file is now imported to new db with all the tables and data.

.sql file is provided in the github repo.

Install python packages to mysql

```
$ pip install mysqlclient mysql-connector-python pymysql
```

Code Modification in the env folder

1)   Navigate to the file - env/lib/python2.7/site-packages/notification/view.py

In this file at line 7 change feed to Feed

```
from django.contrib.syndication.views import Feed
```

2)   Navigate to the file - env/lib/python2.7/site-packages/sx/pisa3/pisa_util.py
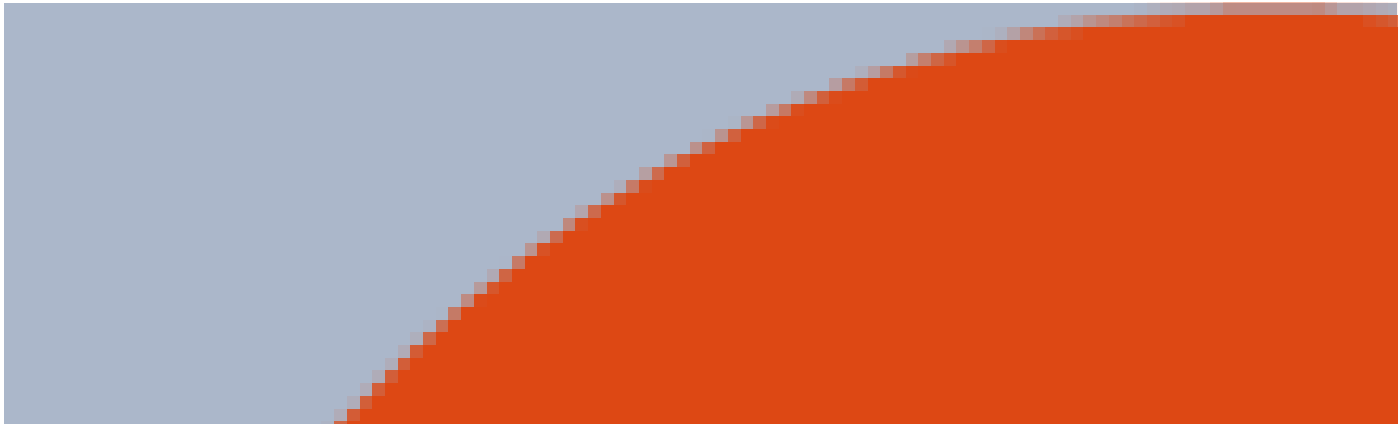
In this file replace lines 55 to 57 with following

```
if not (reportlab.Version[:3]>="2.1"):
    raise ImportError("Reportlab Version 2.1+ is needed!")
REPORTLAB22 = (reportlab.Version[:3]>="2.1")
```

Execute the Application

Edit the local_settings.py and .env file with the database name given and the password.

```
python manage.py runserver --settings=local_settings
```

Serve Beta on Ubuntu

| | Python Installation |

| | Django Setup |

| | Backend Setup |

| | Nginx and Interface Setup |

| | Post Install Steps |

Python Installation

Python:

```
sudo apt install python2.7
```

Since the default python version is 3, change the default python version priority by following

```
$ sudo update-alternatives --install /usr/bin/python python /usr/bin/python2.7 1
$ sudo update-alternatives --install /usr/bin/python python /usr/bin/python3 2
$ sudo update-alternatives --config python
```

Clone Git Repo

```
cd /var
sudo mkdir www
cd www
sudo git clone https://github.com/Sunbird-Serve/serve-beta.git
```

==

Creating and Setting up of Virtual environment

Create virtual environment to install all the necessary applications and dependent libraries within the virtual environment.

```
Check pip version: python -m pip --version

//If pip not installed, execute below commands
curl https://bootstrap.pypa.io/pip/2.7/get-pip.py --output get-pip.py
sudo python2 get-pip.py

//If pip version is not 20.3.4
sudo apt install python-pip 20.3.4
```

Create Virtual Environment

```
cd /var/www/serve-beta
sudo pip install virtualenv
sudo virtualenv env --python=python2.7
cd env
source bin/activate
```

===

Django Setup

==

> *Important: Do not use sudo once virtual env is activated as all the dependent lib need to be installed within the virtual environment.*

==

Install Django

```
$ pip install django==1.4 django-mailer==0.1.0 pip south
```

Setup PEAK rules – required for some of the generic functions used in python.

```
$ cd PEAK-Rules-0.5a1.dev-r2686
$ python setup.py install
$ cd..
//navigate back to previous folder
```

==

Install Remaining Packages

```
$ pip install turbogears xlwt registration notification django-notification==0.2
django-social-auth==0.7.28 Pillow reportlab xhtml2pdf==0.2.4 html5lib==1.0b8 xlwt pisa

$ pip install django-registration==0.8 xlrd python-social-auth==0.2.14
$ pip install pydrive (If version error, then install rsa==4.0) (global)
$ pip install feedparser chronograph simplejson python-dateutil

$ pip install reportlab pypdf2 google-auth-oauthlib boto3~=1.16.57 python-dotenv
$ pip install django-cors-headers==0.11 pyfcm openpyxl redis
```

==

Code Modification in the env folder

1)   Navigate to the file - env/lib/python2.7/site-packages/notification/view.py

In this file at line 7 change feed to Feed

```
from django.contrib.syndication.views import Feed
```

2)   Navigate to the file - env/lib/python2.7/site-packages/sx/pisa3/pisa_util.py

In this file replace lines 55 to 57 with following

```
if not (reportlab.Version[:3]>="2.1"):
    raise ImportError("Reportlab Version 2.1+ is needed!")
REPORTLAB22 = (reportlab.Version[:3]>="2.1")
```

Backend Setup

Install mysql server

```
sudo apt update
(update package index)

sudo apt install mysql-server

systemctl is-active mysql
(To verify installation, should return active)

sudo systemctl start mysql.service
(To start mySQL)
```

===

Change password of the sql server

```
open /etc/mysql/debian.cnf in vi and note down mysql username and password

mysql -u debian-sys-maint -p

sudo mysql
ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password BY 'your_password';
```

===

Import sql file to mysql server

```
mysql -u root -p
create database servebeta;
use servebeta;
source <path to sql filename>;
```

===

Setup Python mysql client connector

```
//Activate the virtual environment
cd /var/www/serve-beta/env

source bin/activate
cd /var/www/serve-beta/evd
----
pip install mysqlclient
sudo apt-get install python2-dev
sudo apt-get install libffi-dev
sudo apt-get install mysql-client
sudo apt-get install libmysqlclient
pip install mysql-connector-python
pip install pymysql
```

===

Nginx and Interface Setup

Nginx Installation

```
sudo apt-get install nginx
sudo uwf app list
sudo uwf allow 'nginx http'
sudo uwf allow 'nginx https'
sudo uwf allow 'nginx full'
```

Certification Generation

```
sudo apt install certbot python3-certbot-nginx
update the conf file
sudo ln -s /etc/nginx/sites-available/domain_name /etc/nginx/sites-enabled/vidyasahyog.lotusp
//restart nginx
sudo certbot --nginx -d domain_name
//restart nginx
//update the conf file and restart nginx again

//unlink the default conf file -
sudo unlink /etc/nginx/sites-enables/default
sudo unlink /etc/nginx/sites-available/default
//restart nginx

permissions:
sudo chown -R www-data:www-data /var/www/serve-beta/*
sudo chmod -R 775 /var/www/serve-beta/
```

Post Install Steps

Activate Environment

```
cd /var/www/serve-beta/env
source bin/activate
```

Interface Command

```
uwsgi --touch-reload=/var/www/serve-beta/evd/reload_uwsgi.ini --close-on-exec -s /tmp/uwsgi_ev
```

Set Global sql mode

```
SET GLOBAL sql_mode=(SELECT REPLACE(@@sql_mode,'ONLY_FULL_GROUP_BY',''));
```