Content Repository using Openstack Object Storage

Objective

To build a media storage solution for storing content on On-Premise Private Cloud.

Module Details

Module 1: OpenStack Installation

OpenStack is a free and open-source software platform for cloud computing, mostly deployed as infrastructure-as-a-service (IaaS), whereby virtual servers and other resources are made available to customers. The software platform consists of interrelated components that control diverse, multi-vendor hardware pools of processing, storage, and networking resources throughout a data center. Users either manage it through a web-based dashboard, through command-line tools or through RESTful web services.

Module 2: Web interface for storage access

Django is a free and open-source web framework, written in Python, which follows the model-view-template (MVT) architectural pattern. Django's primary goal is to ease the creation of complex, database-driven websites. Python is used throughout, even for settings files and data models. Django also provides an optional administrative create, read, update, and delete interface that is generated dynamically through introspection and configured via admin models. We will use Django to build a web interface, to provide authentication and authorization to access the storage and its resource.

Module 3: Object Storage

Object storage (also known as **object-based storage**) is a computer data storage architecture that manages data as objects, as opposed to other storage architectures like file systems that manage data as a file hierarchy and block storage which manages data as blocks within sectors and tracks. Each object typically includes the data itself, a variable amount of metadata, and a globally unique identifier.

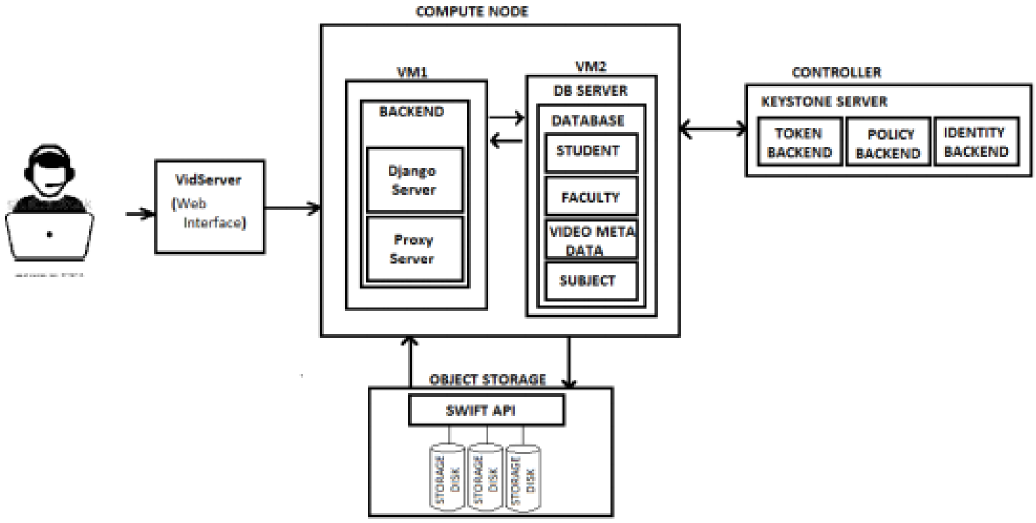
Object-storage systems allow the retention of massive amounts of unstructured data. Object storage is used for purposes such as storing photos on Facebook, songs on Spotify, or files in online collaboration services Dropbox. Hence it is a good option for storage architecture for our project.

Swift

The OpenStack Object Store project, known as Swift, offers cloud storage software so that you can store and retrieve lots of data with a simple API. It's built for scale and optimized for durability, availability, and concurrency across the entire data set. Swift is ideal for storing unstructured data that can grow without bound.

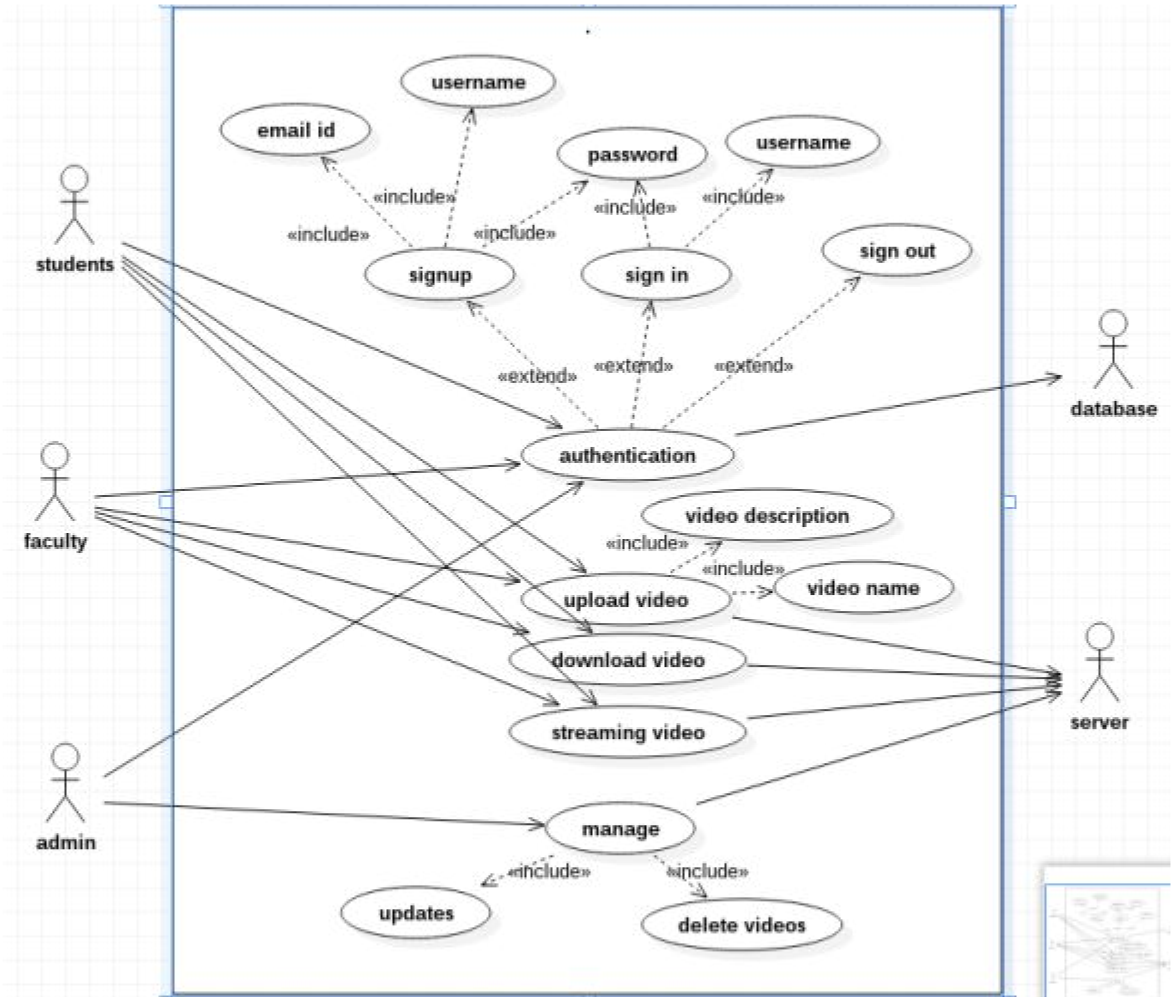
Diagrams

Architecture Diagram

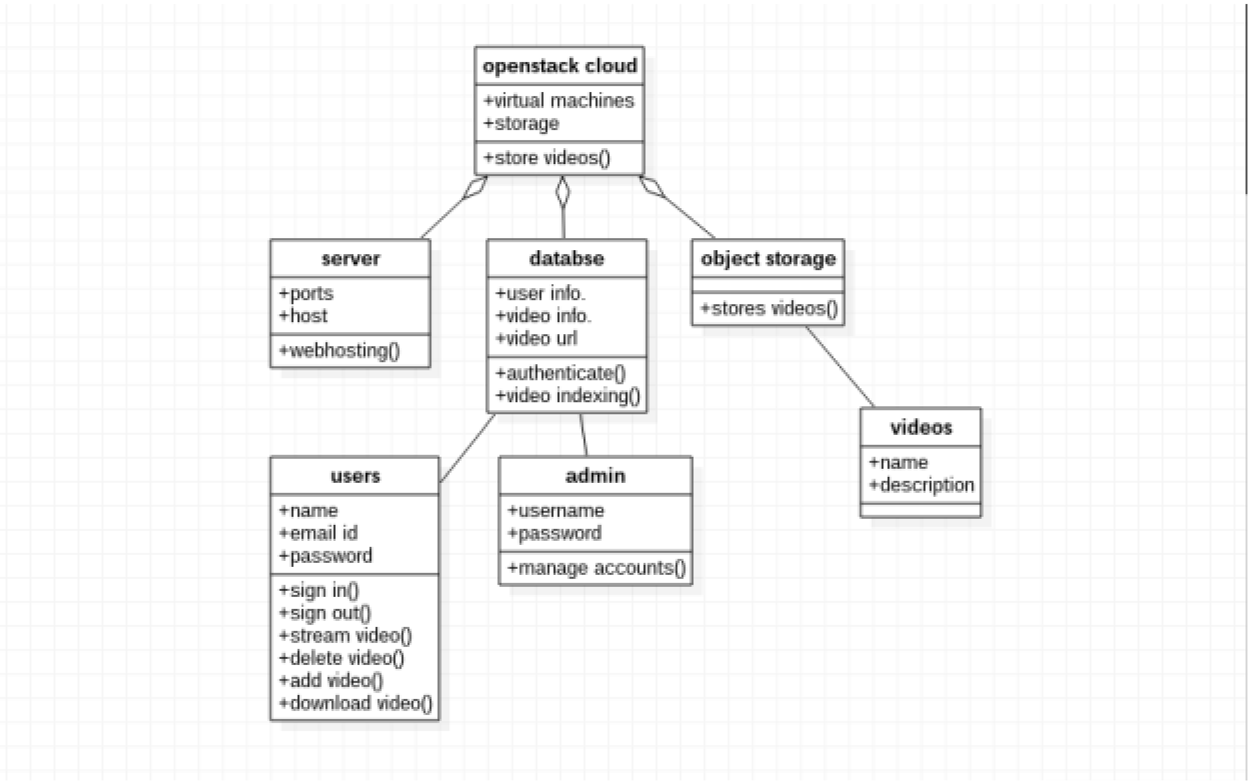


UML Diagrams

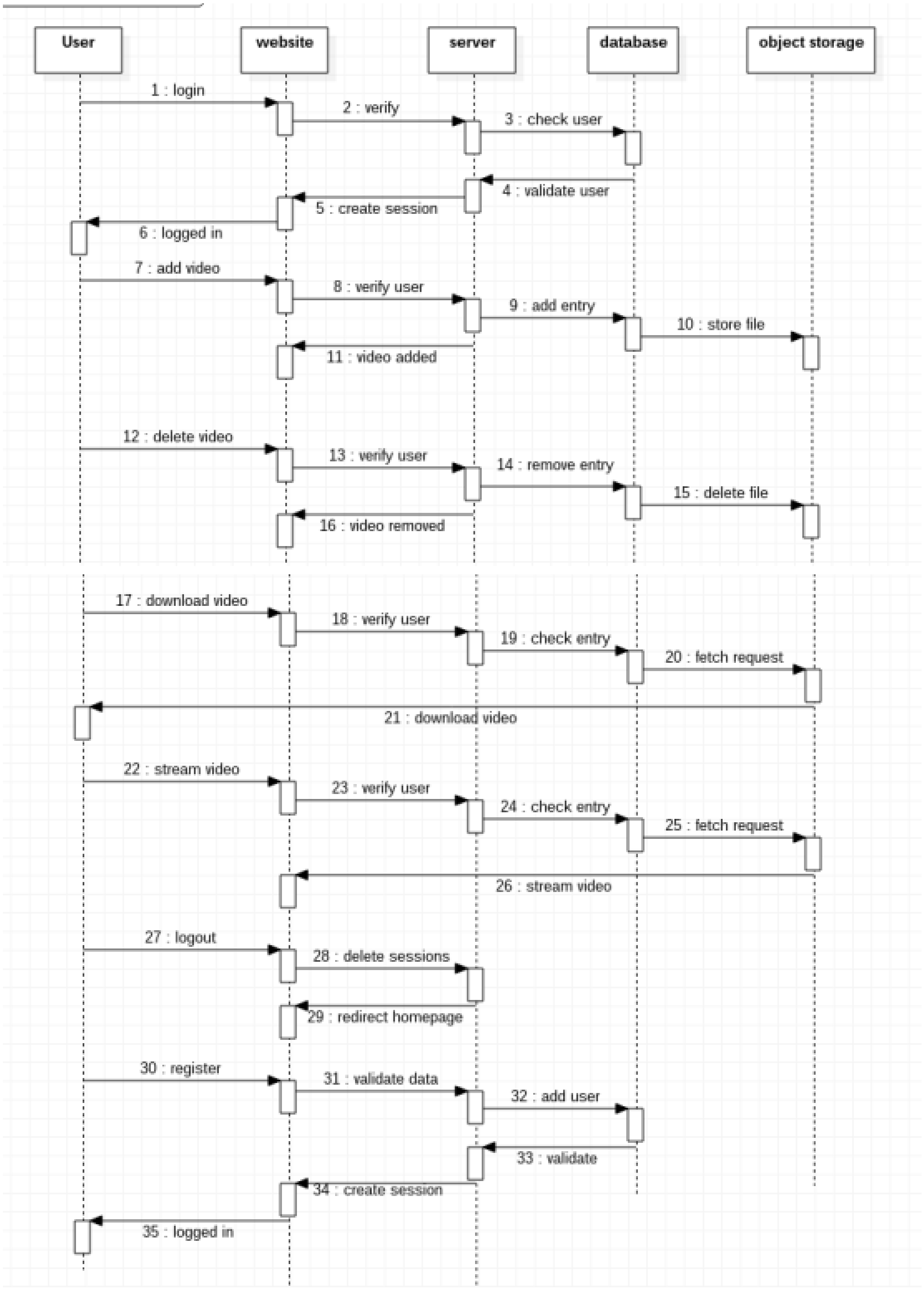
Use case diagram



Class diagram



Sequence diagram



Implementation

Module 1: OpenStack Installation

Installed OpenStack private cloud on a virtual machine.
OpenStack release: Newton

Installed Core services:

1. Identity service (Keystone)
2. Image service (glance)
3. Compute service (Nova)
4. Networking service (neutron)
5. Dashboard (horizon)

Installed additional services:

1. Object Storage services (swift)

Module 2: Web interface for storage access

Created a web interface using the Django web framework and MySQL database to perform CRUD operations on user and records.
Implemented functionalities:

1. SignUp: This is the page where new users can register themselves to use our services.
2. SignIn: To authorize already registered users and create session.
3. Logout: To delete session and logout user.
4. View: To show video catalog and also upload, download, or remove videos.
5. Database: To store information about a user and video metadata.

Module 3: Object Storage

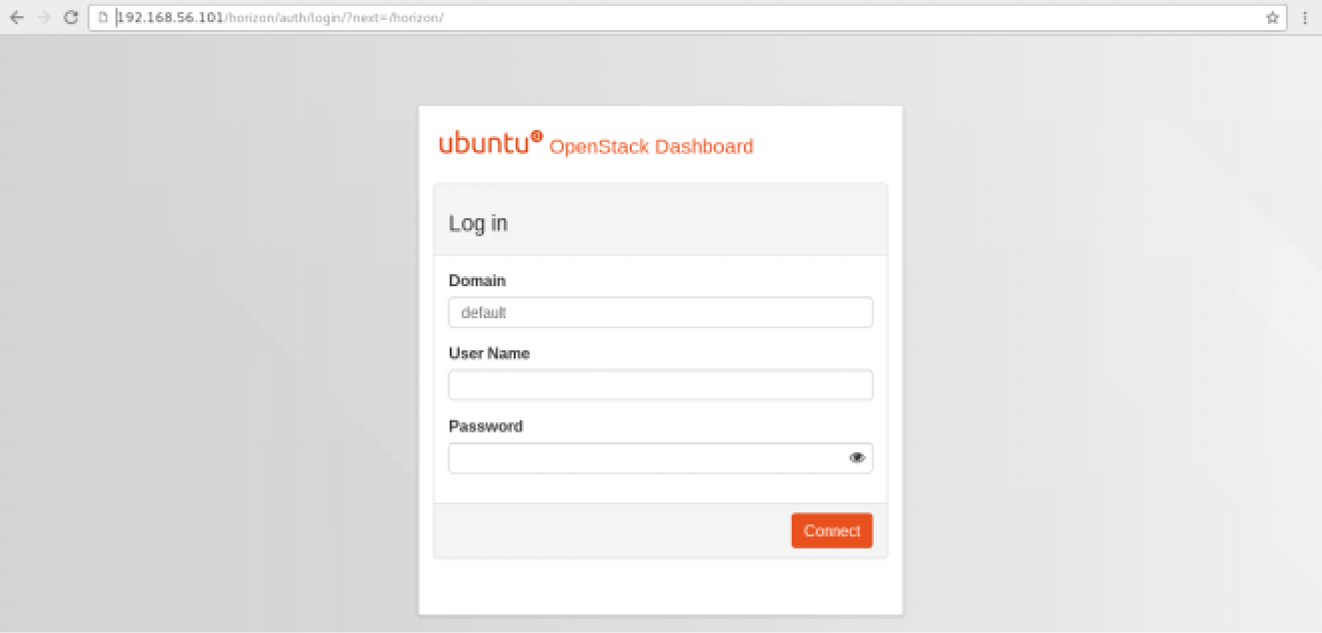
Integrated the web interface with authentication service and object storage service.

Configurations (settings.py):

```
DEFAULT_FILE_STORAGE = 'swift.storage.SwiftStorage'
SWIFT_AUTH_URL='http://controller:5000/v3'
SWIFT_AUTH_VERSION=3
SWIFT_USERNAME='demo'
SWIFT_KEY='linux'
SWIFT_TENANT_NAME='demo'
SWIFT_USER_DOMAIN_NAME='Default'
SWIFT_PROJECT_DOMAIN_NAME='Default'
```

Screenshots

Dashboard


192.168.56.101/horizon/auth/login/?next=/horizon/
ubuntu® OpenStack Dashboard
Connect

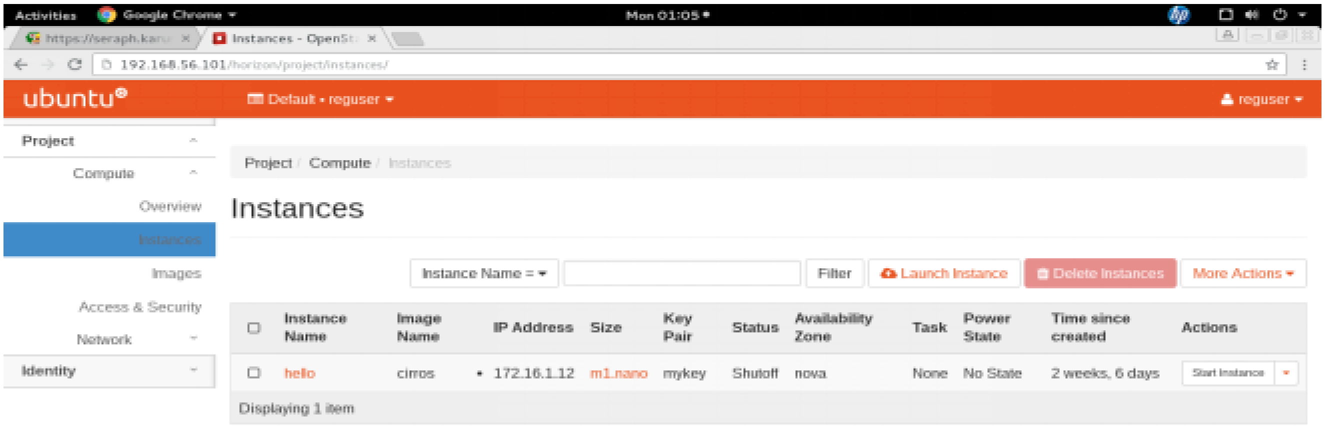
Log in

Domain: default

User Name:

Password:

Instance launched



Project / Compute / Instances

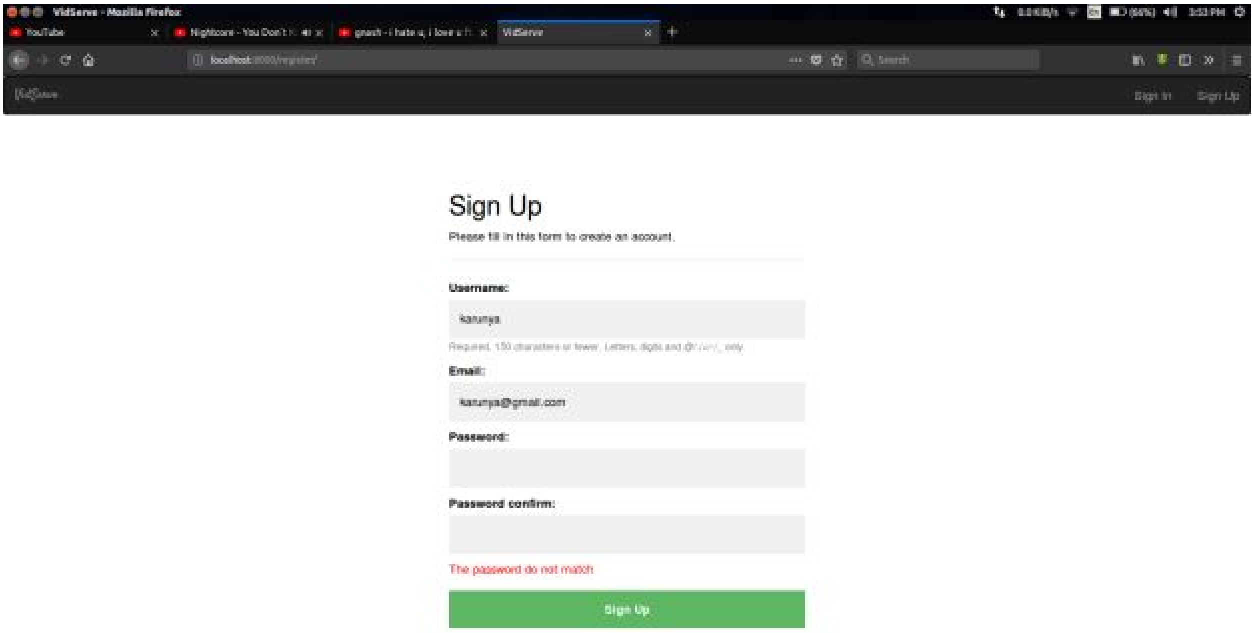
Instances

□	Instance Name	Image Name	IP Address	Size	Key Pair	Status	Availability Zone	Task	Power State	Time since created	Actions
□	hello	cirros	• 172.16.1.12	m1.nano	mykey	Shutoff	nova	None	No State	2 weeks, 6 days	Start Instance

Displaying 1 item

Web Interface

SignUp Page



Sign Up

Please fill in this form to create an account.

Username:

karunya

Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.

Email:

karunya@gmail.com

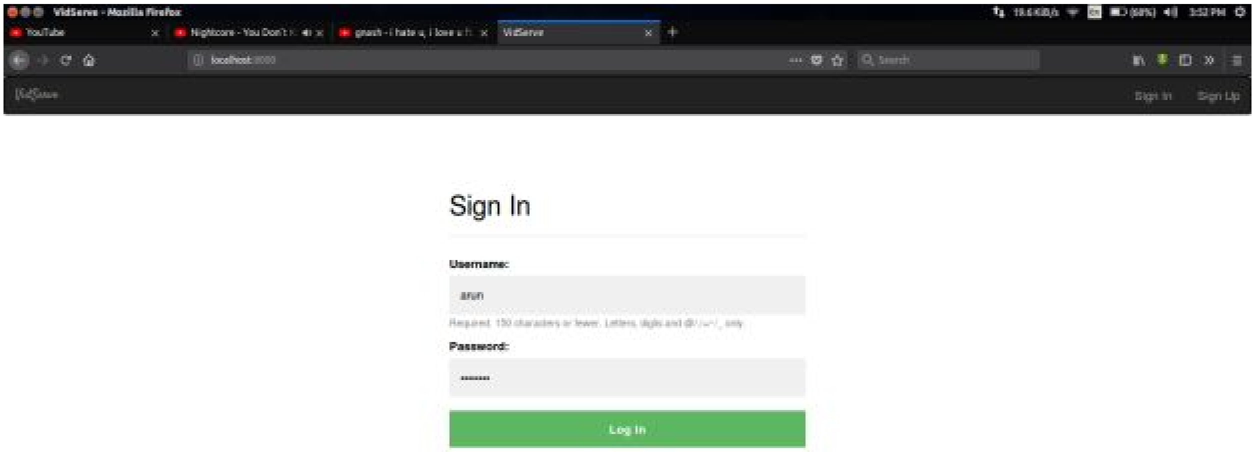
Password:

Password confirm:

The password do not match

Sign Up

SignIn Page



Sign In

Username:

arun

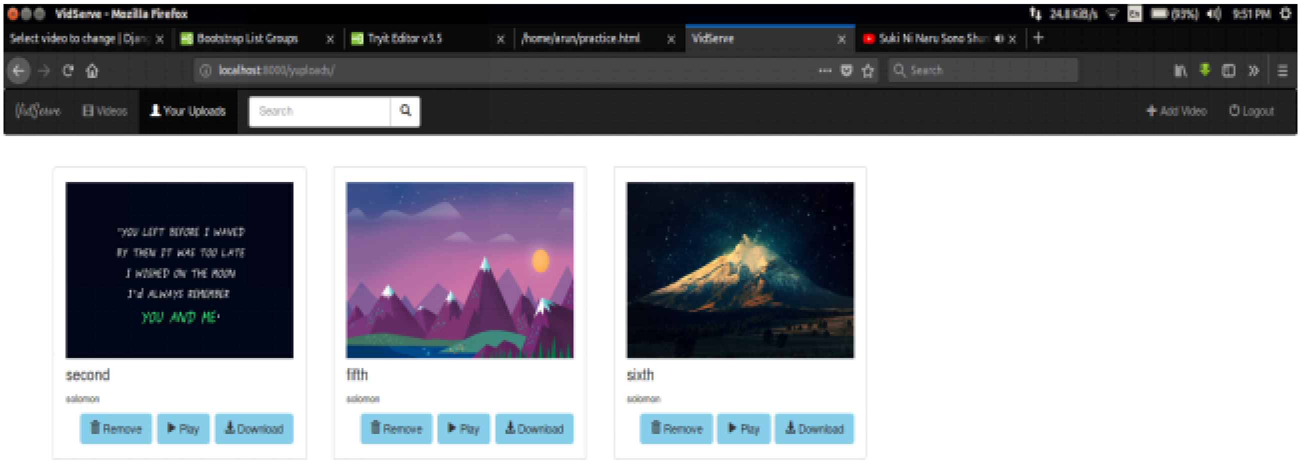
Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.

Password:

••••••••

Log In

Website Dashboard


VidServe - Mozilla Firefox
24.8 KiB/s
(93%)
9:51 PM
Select video to change | Djang
Bootstrap List Groups
Tryit Editor v3.5
/home/arun/practice.html
VidServe
Suki Ni Naru Sono Shun
localhost:8000/uploads/
Search
VidServe
Videos
Your Uploads
Search
Add Video
Logout
"YOU LEFT BEFORE I WAVED
BY THEN IT WAS TOO LATE
I WISHED ON THE MOON
I'd ALWAYS REMEMBER
YOU AND ME"
second
salomon
Remove
Play
Download
fifth
salomon
Remove
Play
Download
sixth
salomon
Remove
Play
Download