

# Development Tools

- MiniO
  - Setup MiniO as AWS S3 alternative in development with docker
- Expose
  - Setup Expose

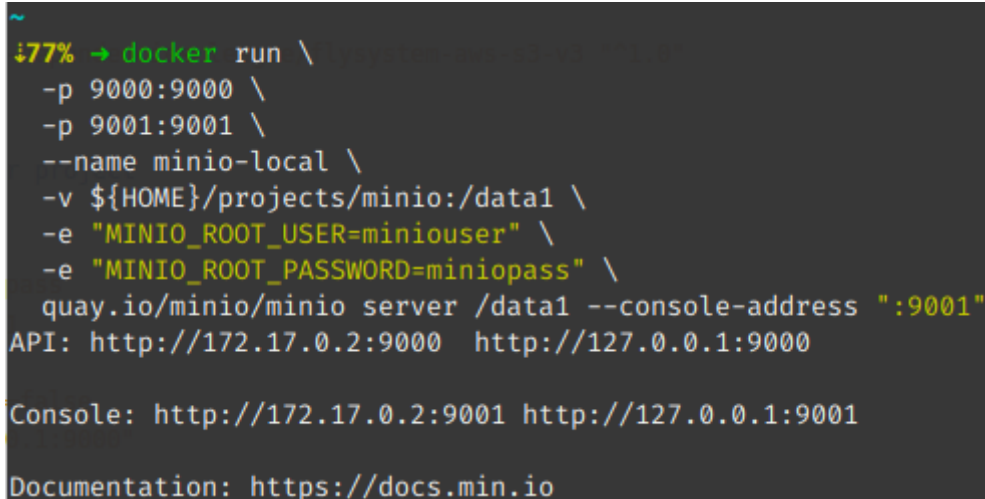
# MiniO

# Setup MinIO as AWS S3 alternative in development with docker

## Installation

1. Make sure your local machine already have Docker installed.
2. Run MinIO:

```
docker run \  
  -p 9000:9000 \  
  -p 9001:9001 \  
  --name minio-local \  
  -v ${HOME}/projects/minio:/data1 \  
  -e "MINIO_ROOT_USER=miniouser" \  
  -e "MINIO_ROOT_PASSWORD=miniopass" \  
  quay.io/minio/minio server /data1 --console-address ":9001"
```



```
~  
↓77% → docker run \  
  -p 9000:9000 \  
  -p 9001:9001 \  
  --name minio-local \  
  -v ${HOME}/projects/minio:/data1 \  
  -e "MINIO_ROOT_USER=miniouser" \  
  -e "MINIO_ROOT_PASSWORD=miniopass" \  
  quay.io/minio/minio server /data1 --console-address ":9001"  
API: http://172.17.0.2:9000 http://127.0.0.1:9000  
  
Console: http://172.17.0.2:9001 http://127.0.0.1:9001  
  
Documentation: https://docs.min.io
```

3. On your Laravel project, install the dependency

```
composer require --with-all-dependencies league/flysystem-aws-s3-v3 "^1.0"
```

#### 4. Change the `.env` on your project

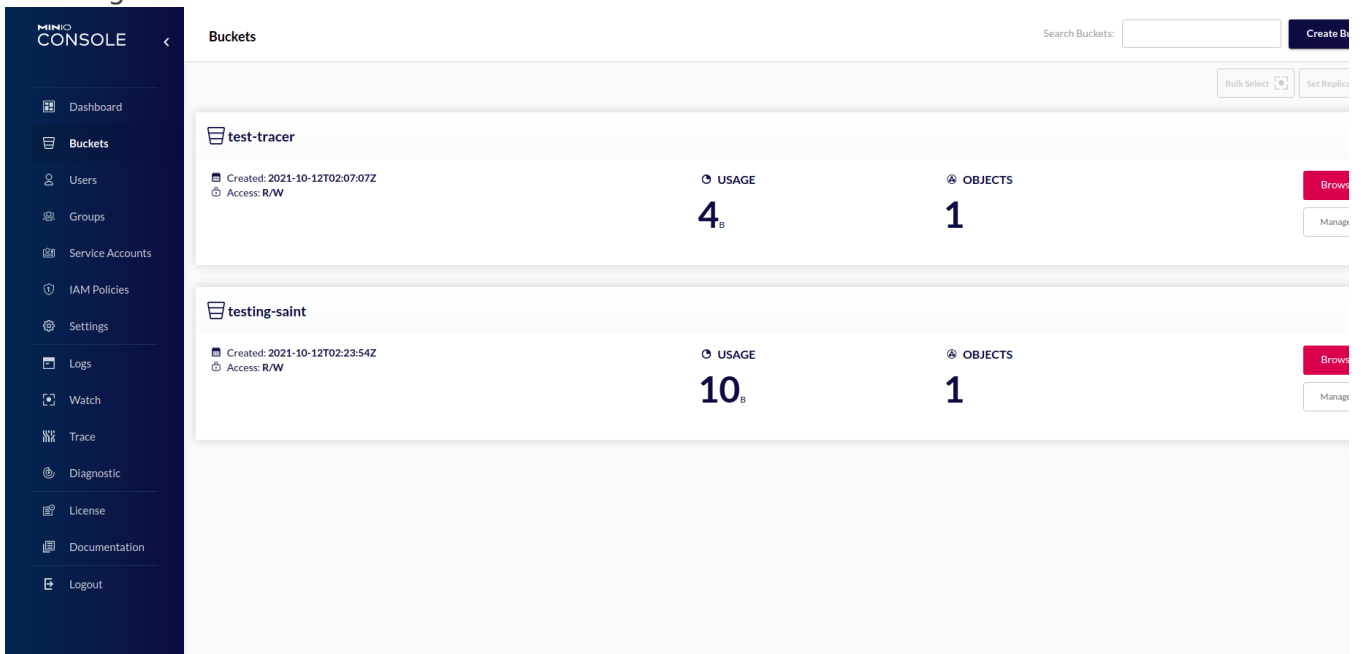
```
AWS_ACCESS_KEY_ID=miniouser
AWS_SECRET_ACCESS_KEY=miniopass
AWS_DEFAULT_REGION=us-east-1
AWS_BUCKET=testing-saint
AWS_USE_PATH_STYLE_ENDPOINT=false
AWS_ENDPOINT="http://127.0.0.1:9000"
```

#### 5. Running after rebooting your PC / Laptop / Mac (shall I mention rpi too?)

```
docker start minio-local
```

#### 6. Accessing the web interface Open your web browser, access `http://localhost:9001`

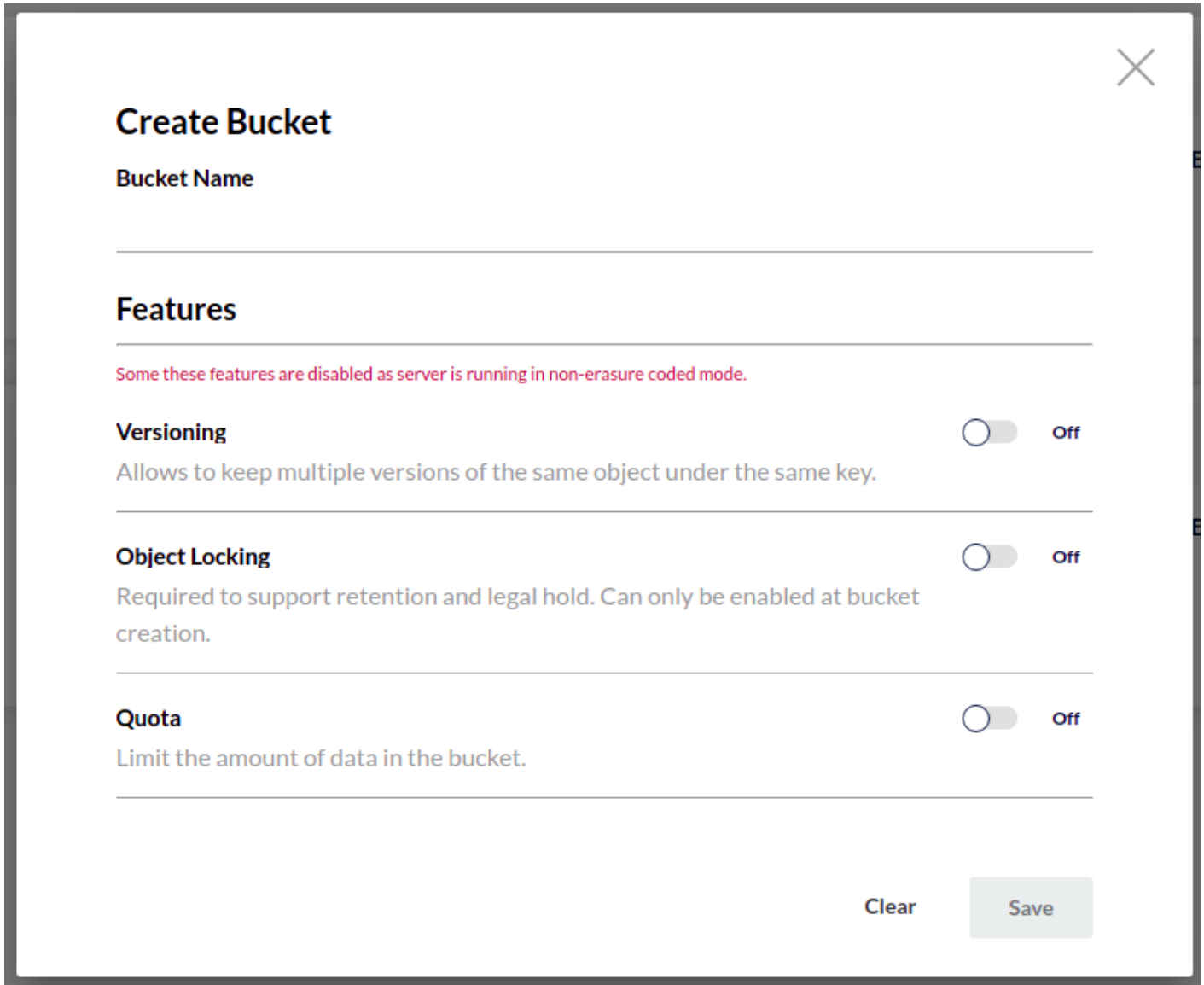
#### 7. Creating bucket



The screenshot displays the Minio Console interface. On the left is a dark sidebar with navigation options: Dashboard, Buckets, Users, Groups, Service Accounts, IAM Policies, Settings, Logs, Watch, Trace, Diagnostic, License, Documentation, and Logout. The main content area is titled 'Buckets' and features a search bar and a 'Create B...' button. Below this, two buckets are listed in a table:

Bucket Name	Created	Access	USAGE	OBJECTS	Actions
test-tracer	2021-10-12T02:07:07Z	R/W	4 <sub>B</sub>	1	Browse, Manage
testing-saint	2021-10-12T02:23:54Z	R/W	10 <sub>B</sub>	1	Browse, Manage

Click 'Create Bucket', and a modal window will be shown like this:



**Create Bucket**

Bucket Name

---

**Features**

---

Some these features are disabled as server is running in non-erasure coded mode.

**Versioning**  Off  
Allows to keep multiple versions of the same object under the same key.

---

**Object Locking**  Off  
Required to support retention and legal hold. Can only be enabled at bucket creation.

---

**Quota**  Off  
Limit the amount of data in the bucket.

---

Clear Save

Set the bucket name. The bucket name will be used as `AWS_BUCKET` in your app.

## Further Reading

[MinIO Documentation](#)

# Expose

# Setup Expose

## Overview

This documentation covers how-to setting up expose in Yayasan Vitka development environment.

## Prerequisites

You need to set working development server first before using expose.

## Installing Expose

```
$ composer global require "beyondcode/expose": "^1.0"
```

## Configuration

1. Run:

```
$ expose token 00000000-2ee4-4106-91ef-2bc888d66660
```

“ This token is only a dummy token. Please refer to internal documentation for real key.

2. Publish the configuration

```
$ expose publish
```

### 3. Edit configuration

```
$ nano ~/.expose/config.php
```

#### Copy

replace `'host' => 'sharedwithexpose.com'` to `'host' => 'dev.yayasanvitka.id'`

## Running Expose

To test if expose is running, go to your web project directory, then run:

```
$ expose share https://siakad.test --subdomain=siakad
```

#### Copy

This command shares your local `https://siakad.test` to `https://siakad.dev.yayasanvitka.id`. Visit the url to see if it is working.