wai-sdg-portal

Akvo

Sep 07, 2023

INSTALL

CHAPTER

ONE

INSTALLATION GUIDE

Warning: Below step is for production-ready installation process. Please follow Developer-Guide to setup the development mode.

1.1 System Requirements

System Memory
4 GiB

CPU
2 GHz Dual Core Processor

Storage
25 GiB or more Disk

Operating System
Ubuntu Server 22.04

1.2 Prerequisite

Docker Engine
20.10 or above

Git
2.39 or above

3rd Party Service Providers

- Auth0
- Mailjet

1.3 Preparation

Note: The following guide is an example installation on **Ubuntu and Debian based systems**. It has been with **Ubuntu 22.04**.

1.3.1 Install Docker Engine

You need the latest Docker version installed. If you do not have it, please see the following installation guide to get it.

1. Update the apt package index and install packages to allow apt to use a repository over HTTPS:

```
sudo apt update
sudo apt install ca-certificates curl gnupg lsb-release
```

2. Add Docker's official GPG key:

```
sudo mkdir -p /etc/apt/keyrings
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /
↪etc/apt/keyrings/docker.gpg
```

3. Use the following command to set up the repository:

```
echo \
 "deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.gpg]␣
↪https://download.docker.com/linux/ubuntu \
 $(lsb_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.list > /dev/
↪null
```

4. Update the apt package index:

```
sudo apt update
```

5. Install Docker Engine, containerd, and Docker Compose.

```
sudo apt-get install \
 docker-ce docker-ce-cli \
 containerd.io docker-compose-plugin
```

6. Manage Docker as a non-root user

```
sudo groupadd docker
sudo usermod -aG docker $USER
newgrp docker
```

1.3.2 Install Git Version Control

The WAI SDG Portal uses git as version control. Therefore it is better to install git to make it easier to retrieve updates instead download the repository zip.

```
sudo apt install git
```

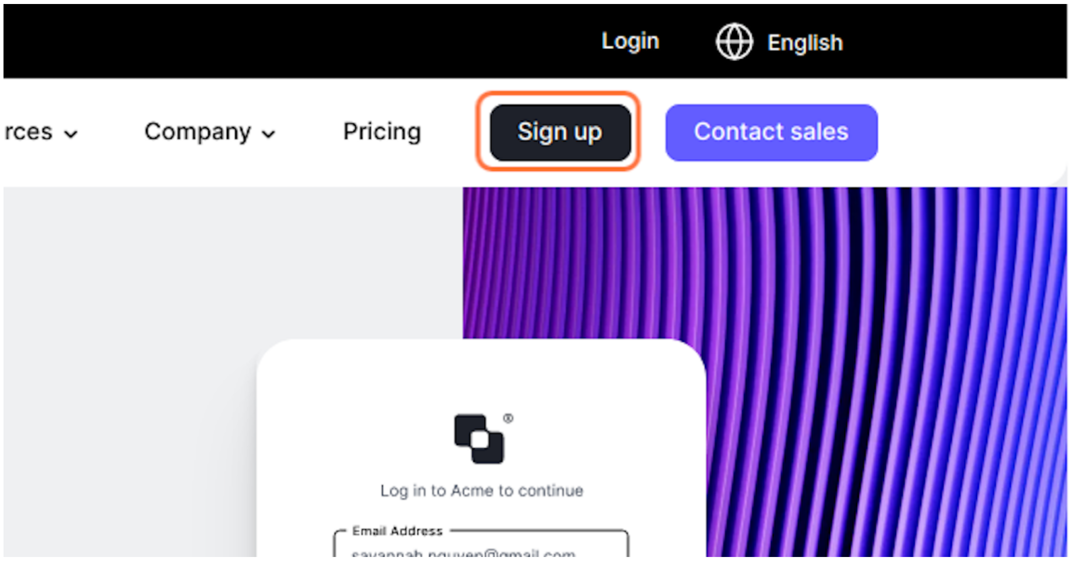
1.3.3 Auth0 Identity Providers

This application **DO NOT** store directly any personal information. WAI SDG Portal uses AUTH0 for a flexible solution to add authentication services.

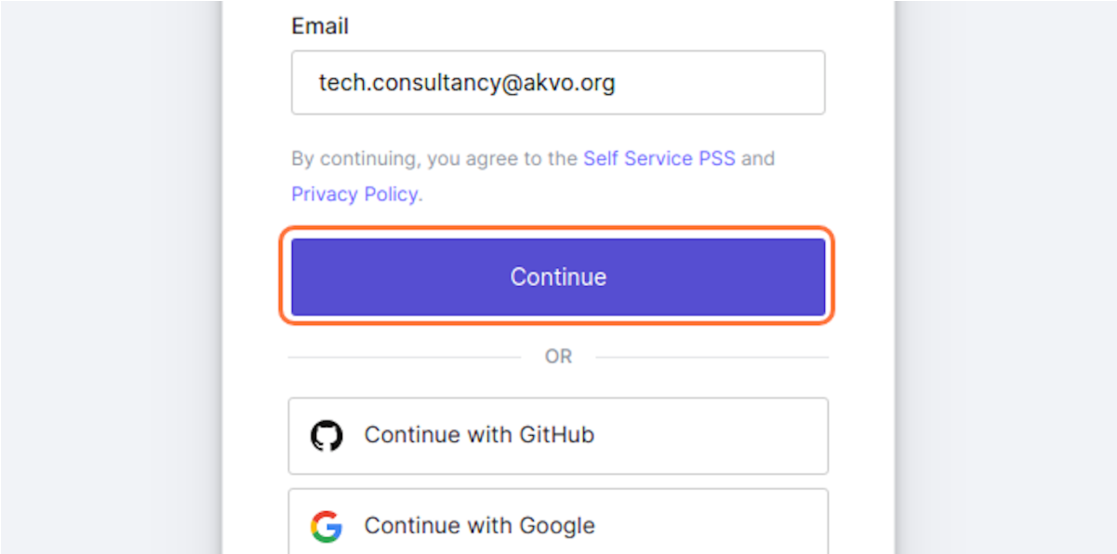
Please visit AUTH0, then follow below guide:

Create Auth0 Account

1. Visit Auth0.com



2. Sign up new account

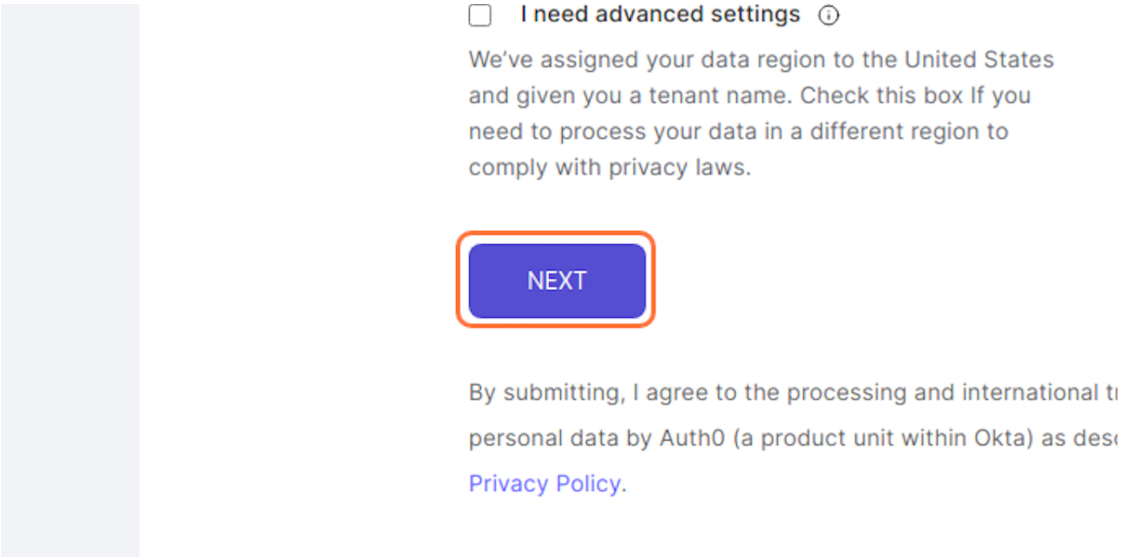


Email

tech.consultancy@akvo.org

By continuing, you agree to the Self Service PSS and Privacy Policy.

3. **Finish Sign Up**



☐ I need advanced settings

We've assigned your data region to the United States and given you a tenant name. Check this box If you need to process your data in a different region to comply with privacy laws.

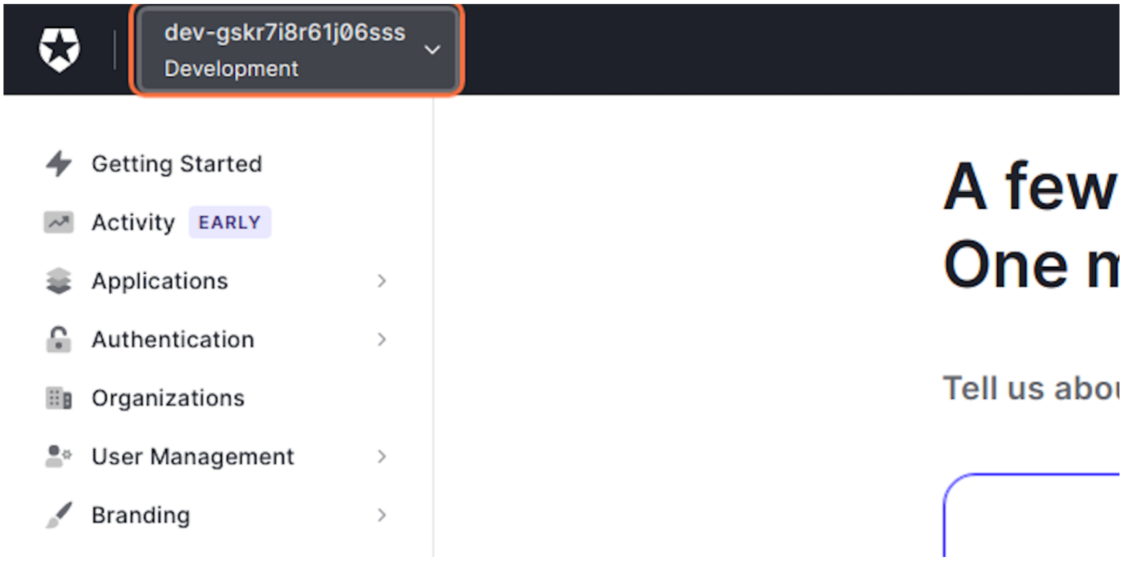
By submitting, I agree to the processing and international t

personal data by Auth0 (a product unit within Okta) as des

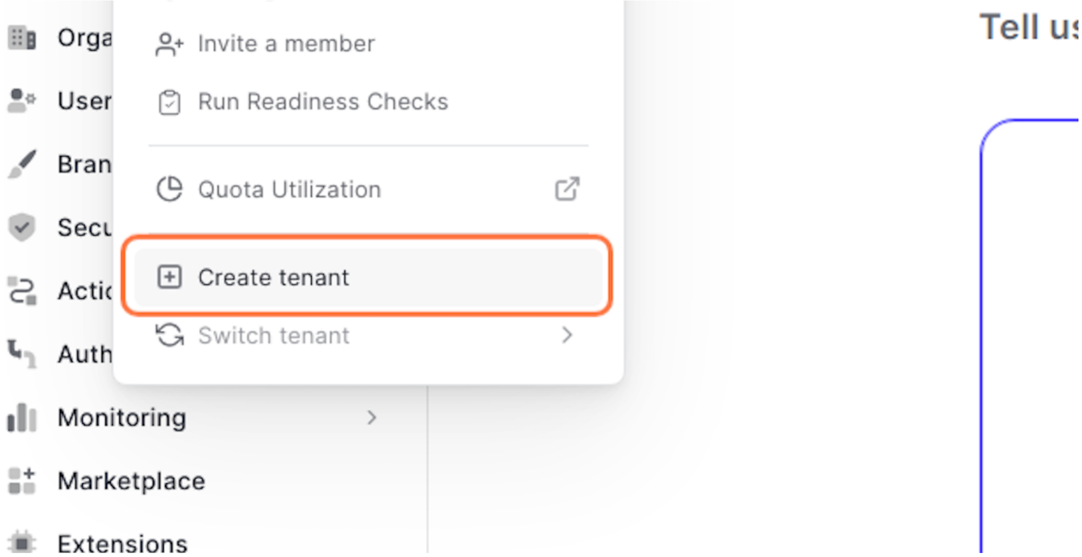
Privacy Policy.

Creating New Tenant

1. **Navigate to Left Corner Dropdown**

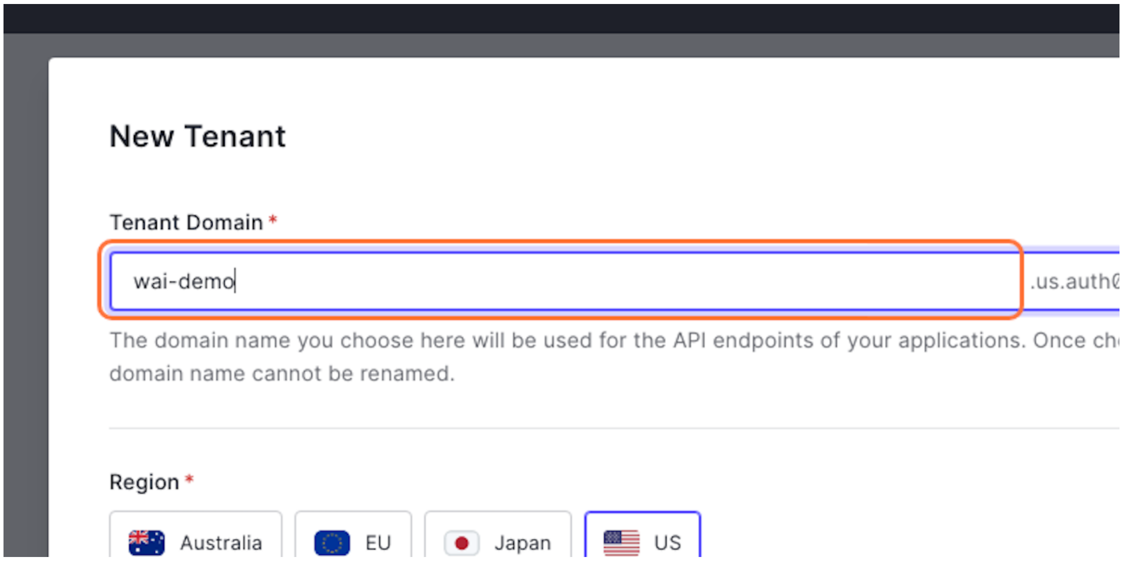


2. **Click create New Tenant**



3. **Fill the Tenant Domain**

Note: Please use your website domain so it can be more consistent



New Tenant

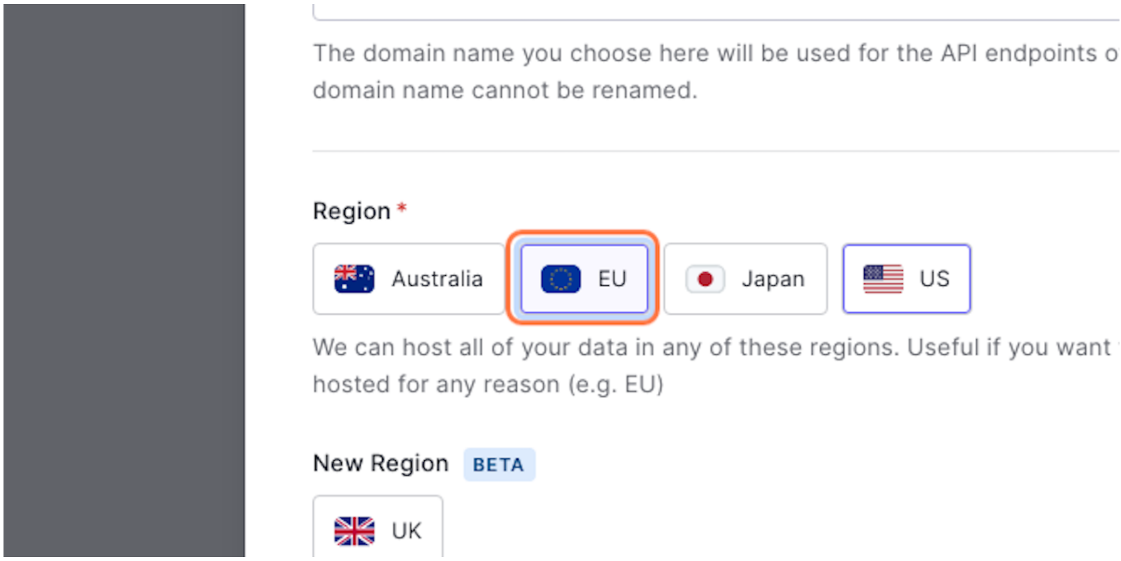
Tenant Domain *

wai-demo .us.auth0

The domain name you choose here will be used for the API endpoints of your applications. Once ch
domain name cannot be renamed.

Region *

4. **Choose region that is closer to your server**



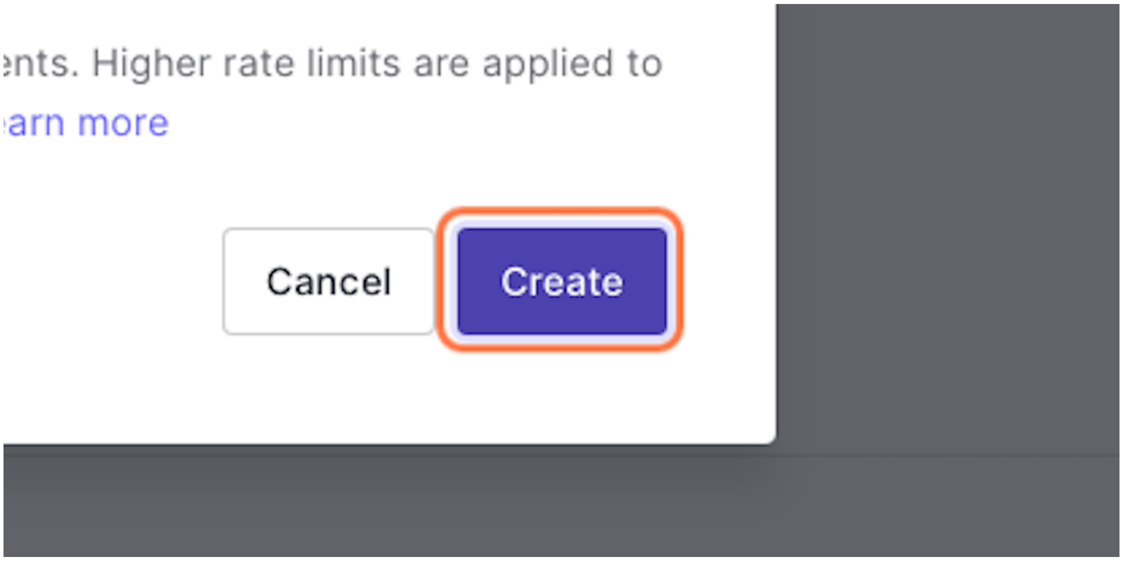
The domain name you choose here will be used for the API endpoints o
domain name cannot be renamed.

Region *

We can host all of your data in any of these regions. Useful if you want
hosted for any reason (e.g. EU)

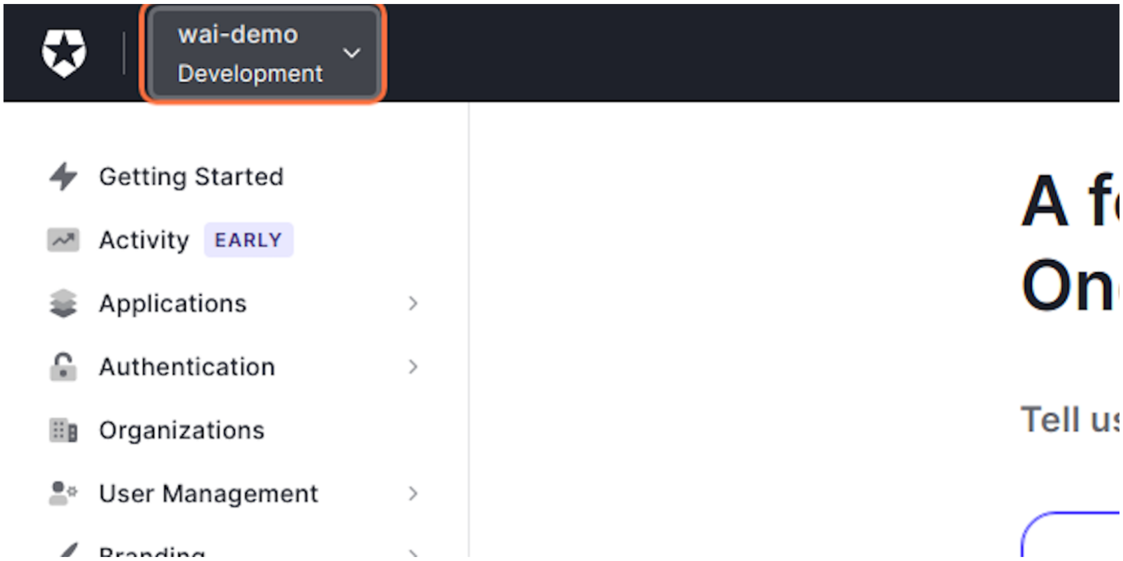
New Region BETA

5. **Finish, Click Create**



ents. Higher rate limits are applied to
arn more

Now you have switched to your new tenant**

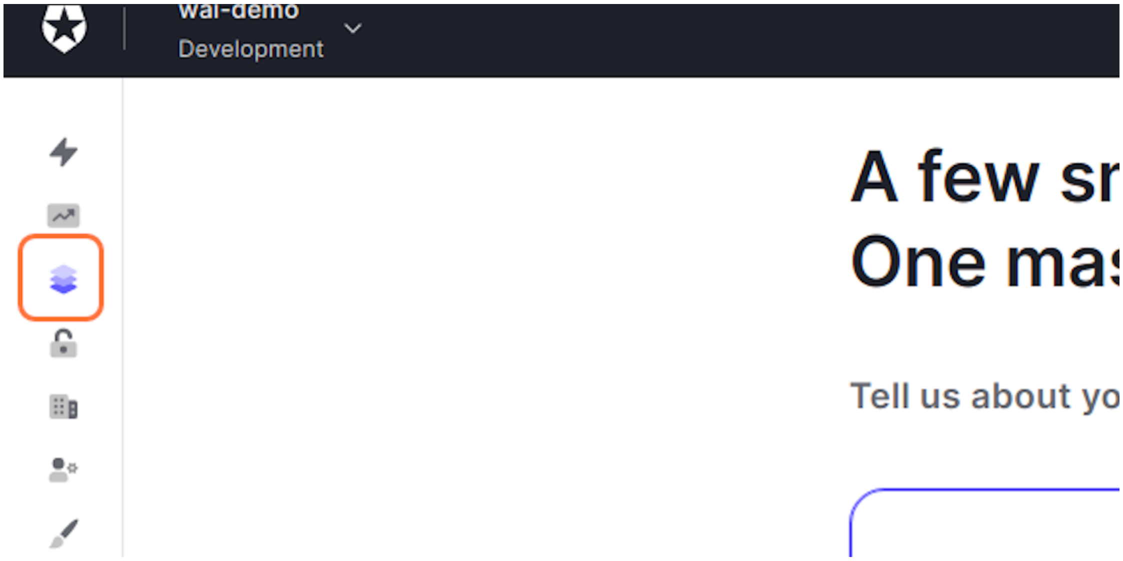


Create Application

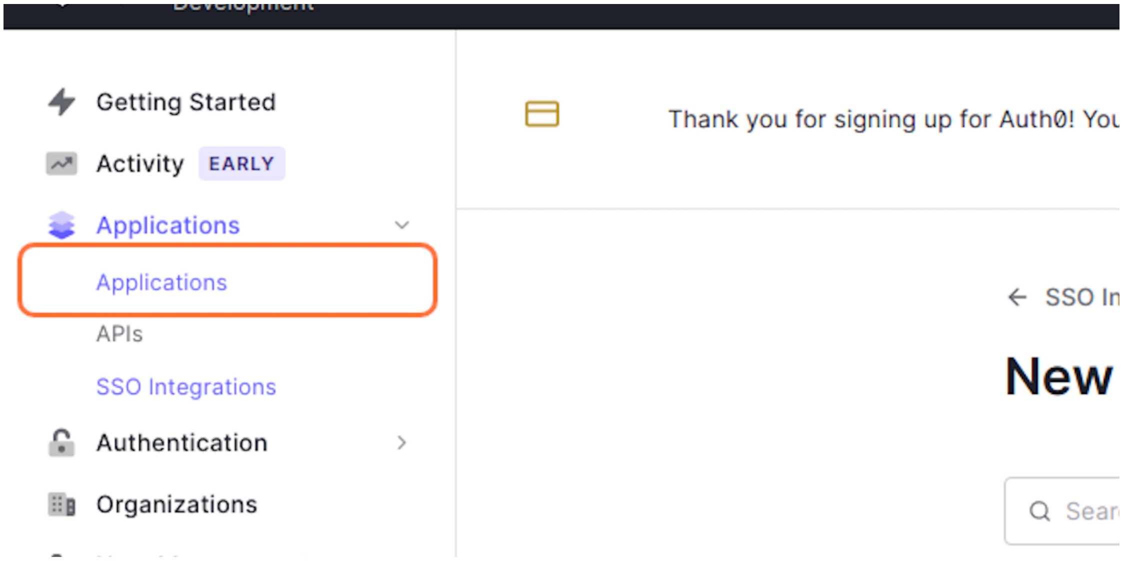
To run this site, 2 types of applications are required for the authorization and authentication process. The first application is the Single Page Application (SPA) type authentication which needed by The JavaScript frontend to authenticate. The second application is for verifying the token that has been created by the frontend which will be used for the process of authorizing access to all the Backend endpoints (API)

Create New SPA Application

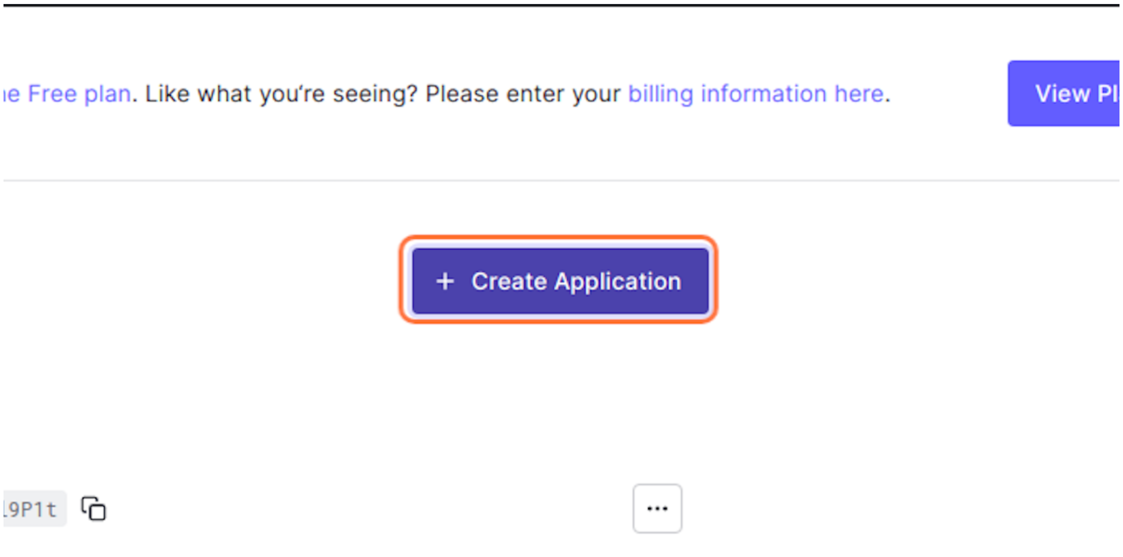
1. Click the 3rd icon in the right corner



2. Click Applications

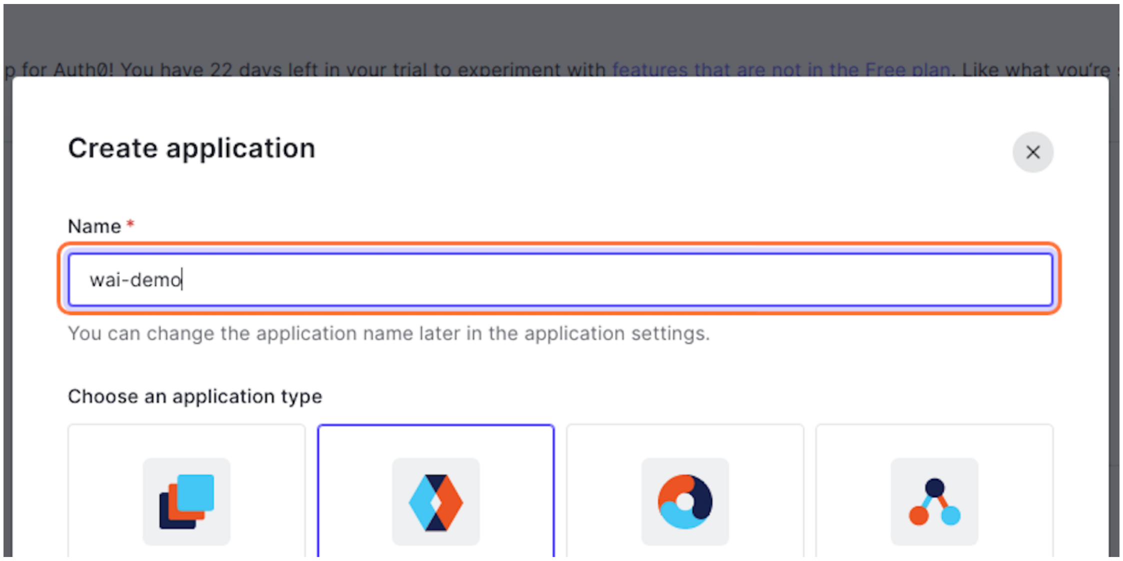


3. Click Create Application



ıe Free plan. Like what you're seeing? Please enter your billing information here.

4. **Fill the name field then Click Create**



p for Auth0! You have 22 days left in your trial to experiment with features that are not in the Free plan. Like what you're

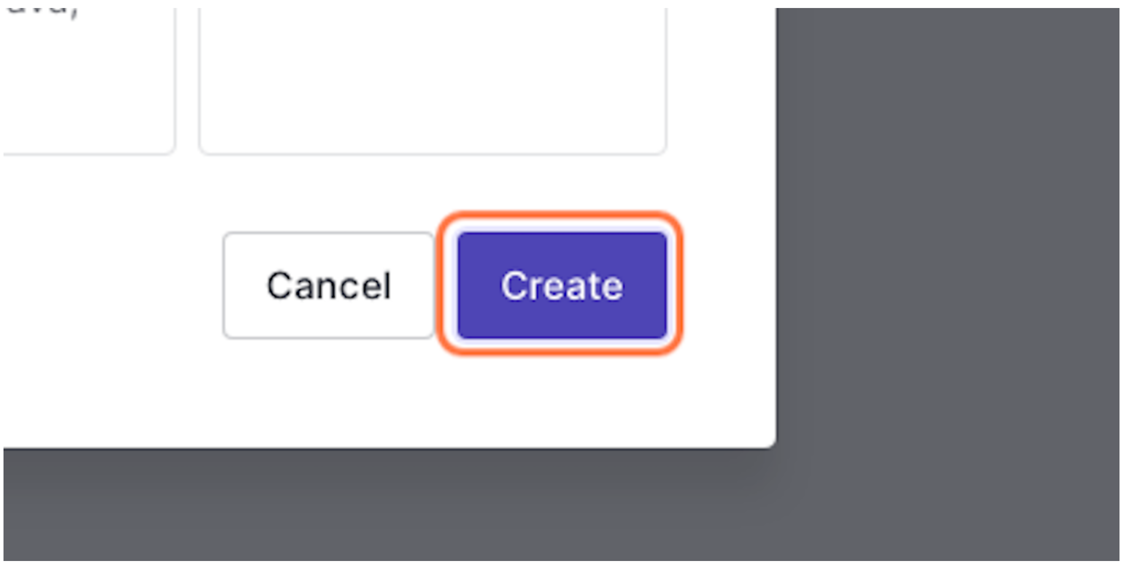
Create application

Name *

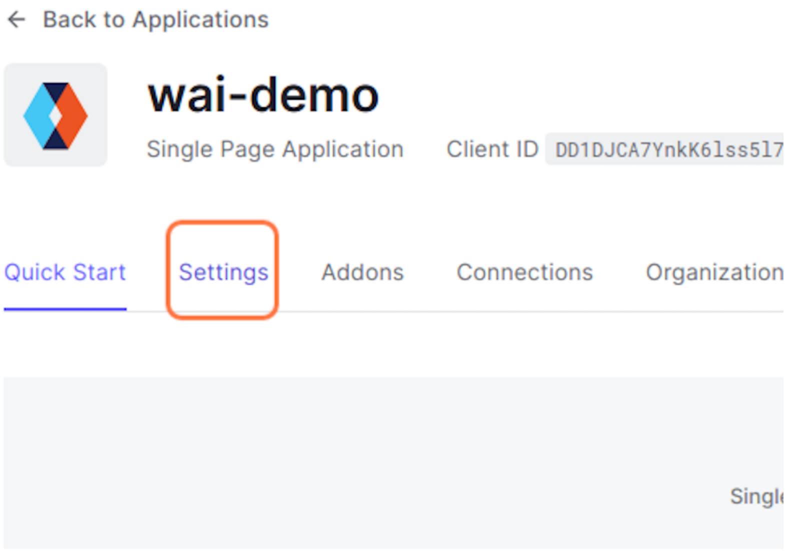
wai-demo

You can change the application name later in the application settings.

Choose an application type



5. **Select Settings Tab**

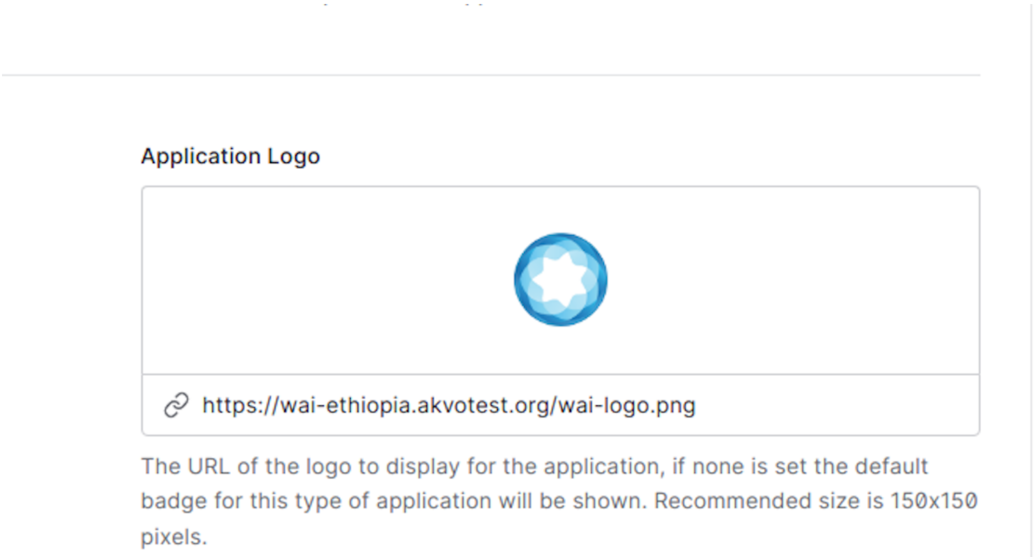


wai-demo

Single Page Application

Client ID DD1DJCA7YnkK6lss517

6. **Add application logo**



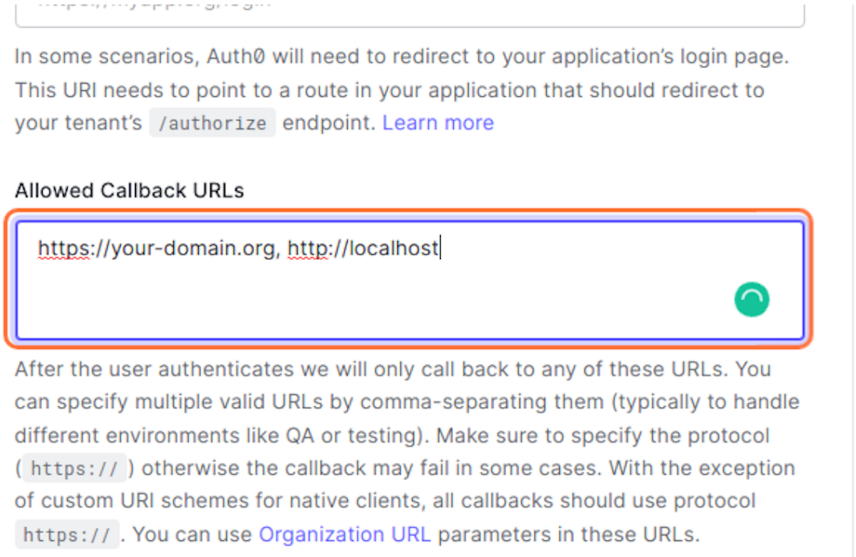
Application Logo

https://wai-ethiopia.akvotest.org/wai-logo.png

The URL of the logo to display for the application, if none is set the default badge for this type of application will be shown. Recommended size is 150x150 pixels.

Note: This application has pre-defined logo, the url of image is available in your installation once this up is up and running in your domain. (eg. https://your-domain.com/wai-logo.png)

7. **Change your-domain.com with your app-domain for all the field below**

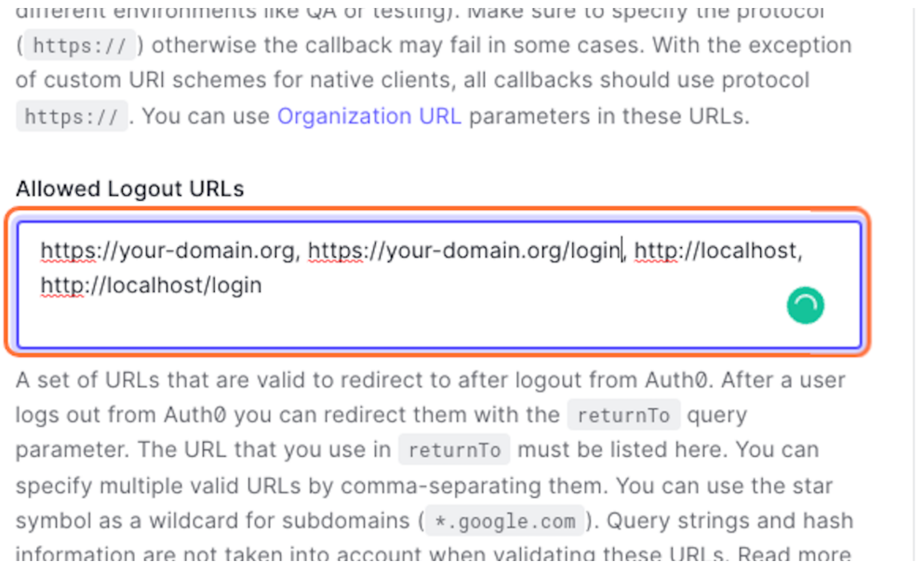


In some scenarios, Auth0 will need to redirect to your application's login page. This URI needs to point to a route in your application that should redirect to your tenant's `/authorize` endpoint. Learn more

Allowed Callback URLs

https://your-domain.org, http://localhost

After the user authenticates we will only call back to any of these URLs. You can specify multiple valid URLs by comma-separating them (typically to handle different environments like QA or testing). Make sure to specify the protocol (`https://`) otherwise the callback may fail in some cases. With the exception of custom URI schemes for native clients, all callbacks should use protocol `https://` . You can use Organization URL parameters in these URLs.

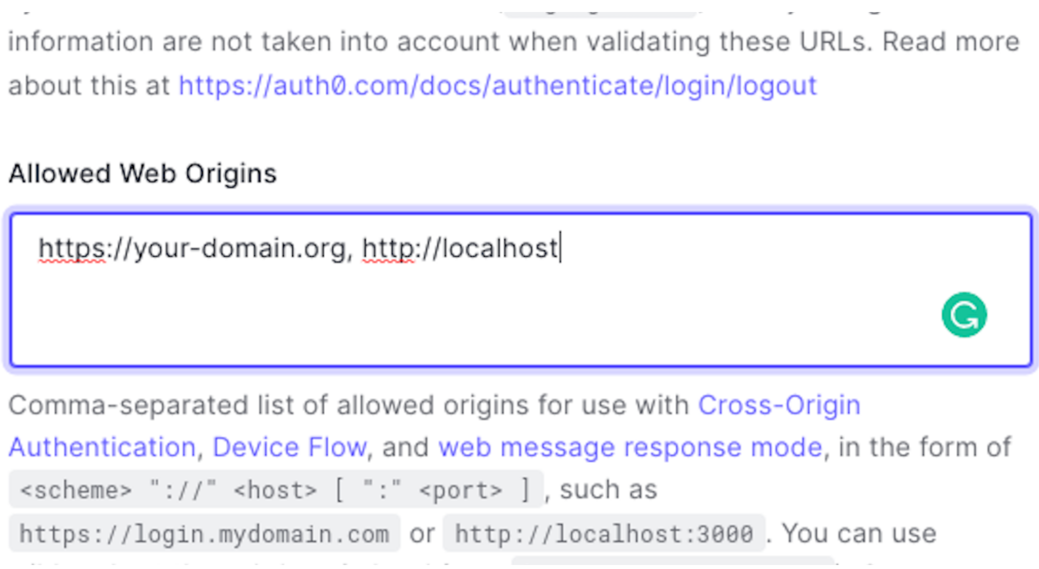


different environments like QA or testing). Make sure to specify the protocol (`https://`) otherwise the callback may fail in some cases. With the exception of custom URI schemes for native clients, all callbacks should use protocol `https://`. You can use Organization URL parameters in these URLs.

Allowed Logout URLs

https://your-domain.org, https://your-domain.org/login, http://localhost, http://localhost/login

A set of URLs that are valid to redirect to after logout from Auth0. After a user logs out from Auth0 you can redirect them with the `returnTo` query parameter. The URL that you use in `returnTo` must be listed here. You can specify multiple valid URLs by comma-separating them. You can use the star symbol as a wildcard for subdomains (`*.google.com`). Query strings and hash information are not taken into account when validating these URLs. Read more

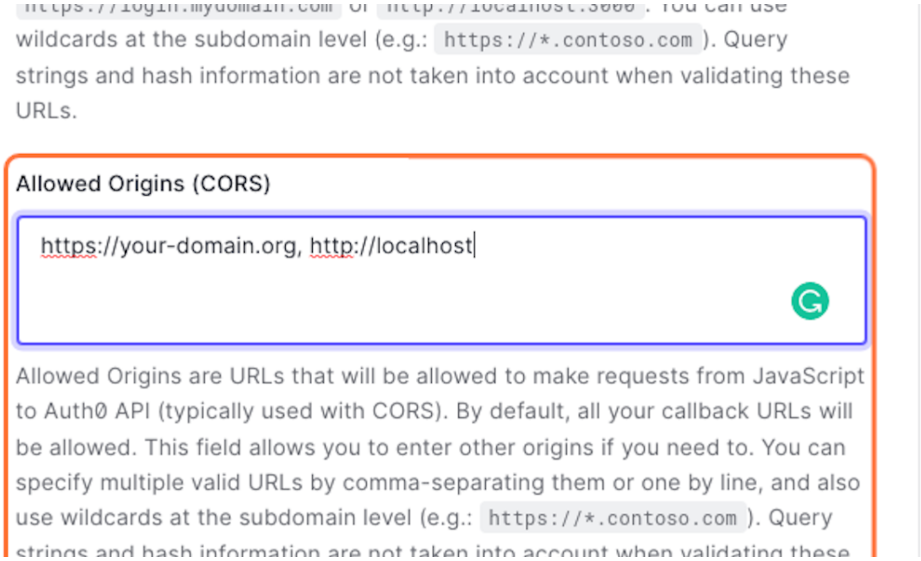


information are not taken into account when validating these URLs. Read more about this at https://auth0.com/docs/authenticate/login/logout

Allowed Web Origins

https://your-domain.org, http://localhost

Comma-separated list of allowed origins for use with Cross-Origin Authentication, Device Flow, and web message response mode, in the form of `<scheme> "://" <host> [ ":" <port> ]`, such as `https://login.mydomain.com` or `http://localhost:3000`. You can use



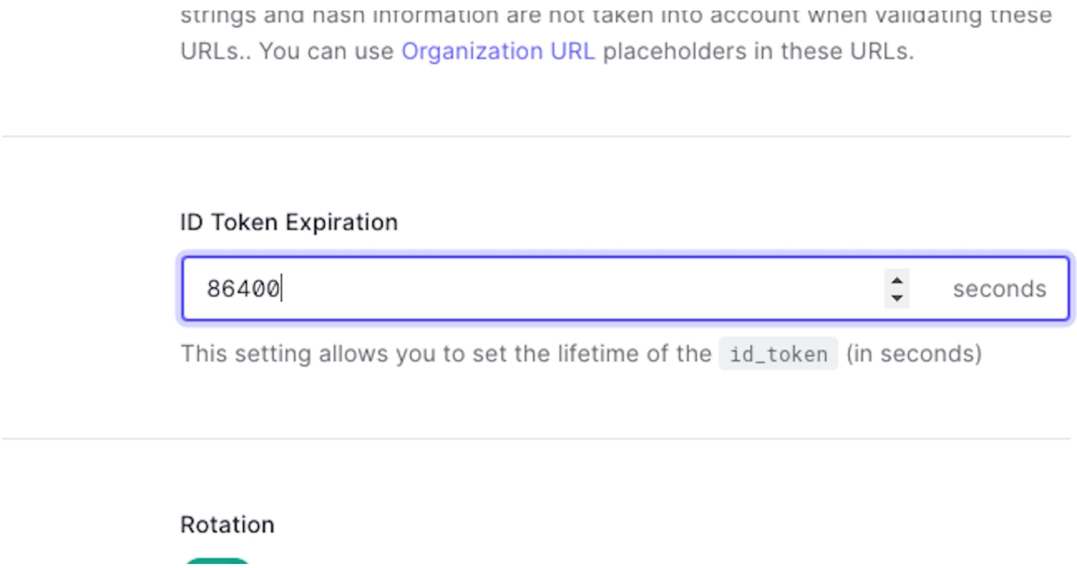
https://login.mydomain.com or http://localhost:3000. You can use wildcards at the subdomain level (e.g.: `https://*.contoso.com`). Query strings and hash information are not taken into account when validating these URLs.

Allowed Origins (CORS)

https://your-domain.org, http://localhost

Allowed Origins are URLs that will be allowed to make requests from JavaScript to Auth0 API (typically used with CORS). By default, all your callback URLs will be allowed. This field allows you to enter other origins if you need to. You can specify multiple valid URLs by comma-separating them or one by line, and also use wildcards at the subdomain level (e.g.: `https://*.contoso.com`). Query strings and hash information are not taken into account when validating these

8. Modify the ID Token Expiration



strings and hash information are not taken into account when validating these URLs.. You can use Organization URL placeholders in these URLs.

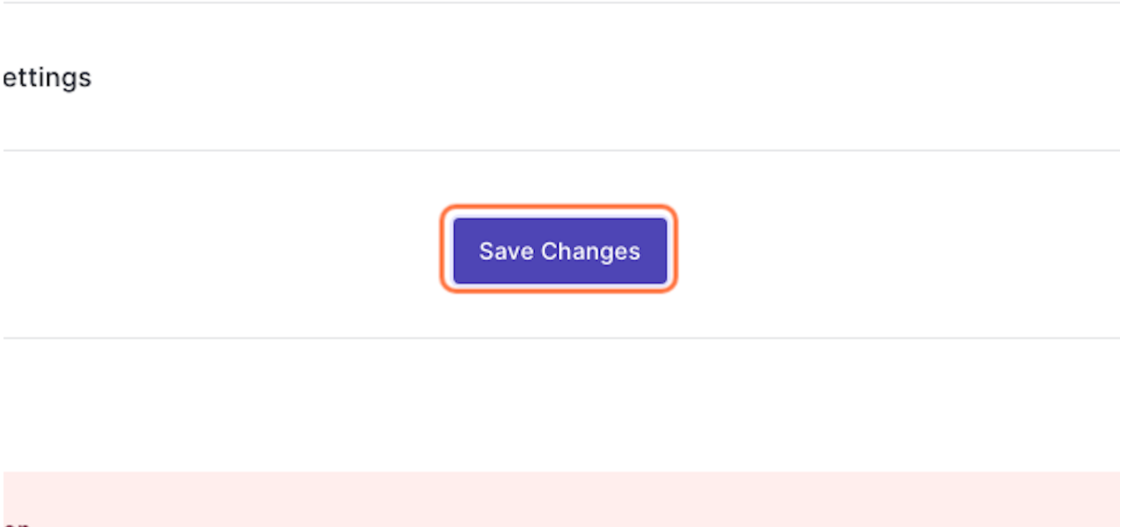
ID Token Expiration

86400 seconds

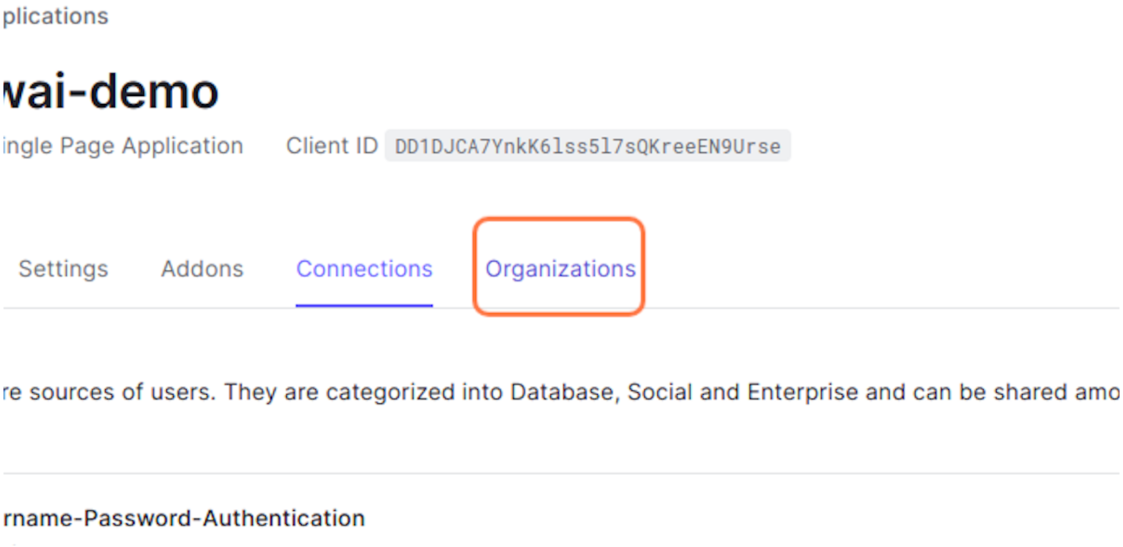
This setting allows you to set the lifetime of the `id_token` (in seconds)

Rotation

9. Click Save Button



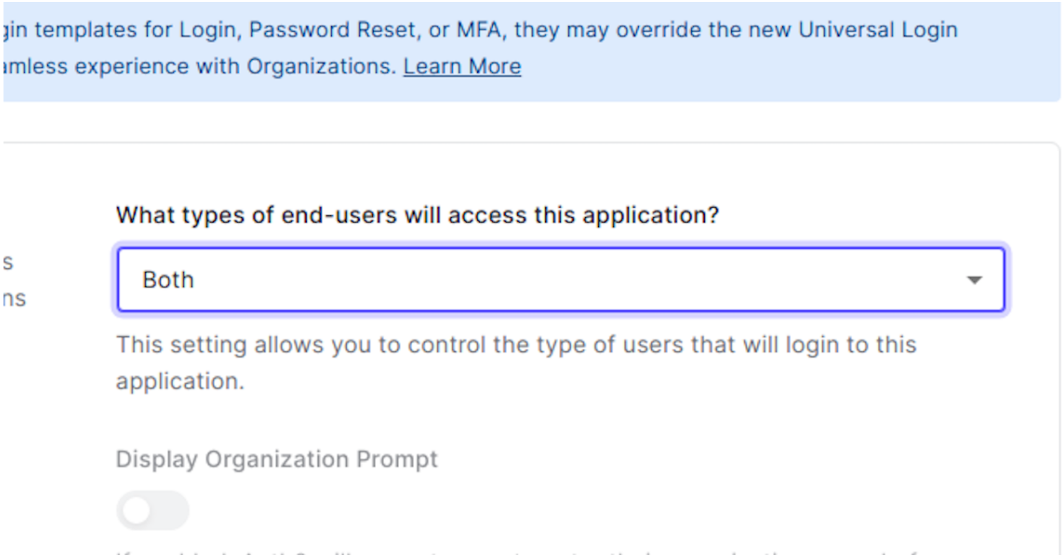
10. **Select Connections Tab**



Client ID DD1DJCA7YnkK6lss5l7sQKreeEN9Urse

re sources of users. They are categorized into Database, Social and Enterprise and can be shared amo

11. **Change below option field with Both**



ɟin templates for Login, Password Reset, or MFA, they may override the new Universal Login ımless experience with Organizations. Learn More

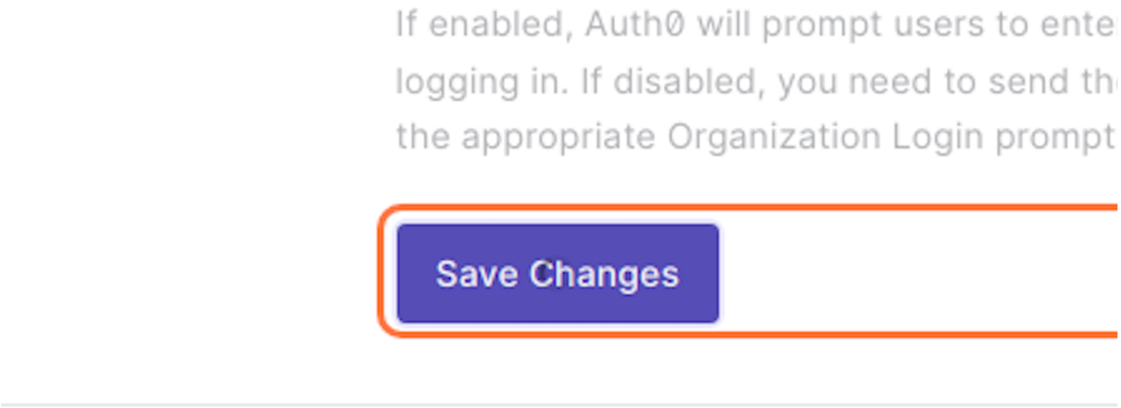
What types of end-users will access this application?

Both

This setting allows you to control the type of users that will login to this application.

Display Organization Prompt

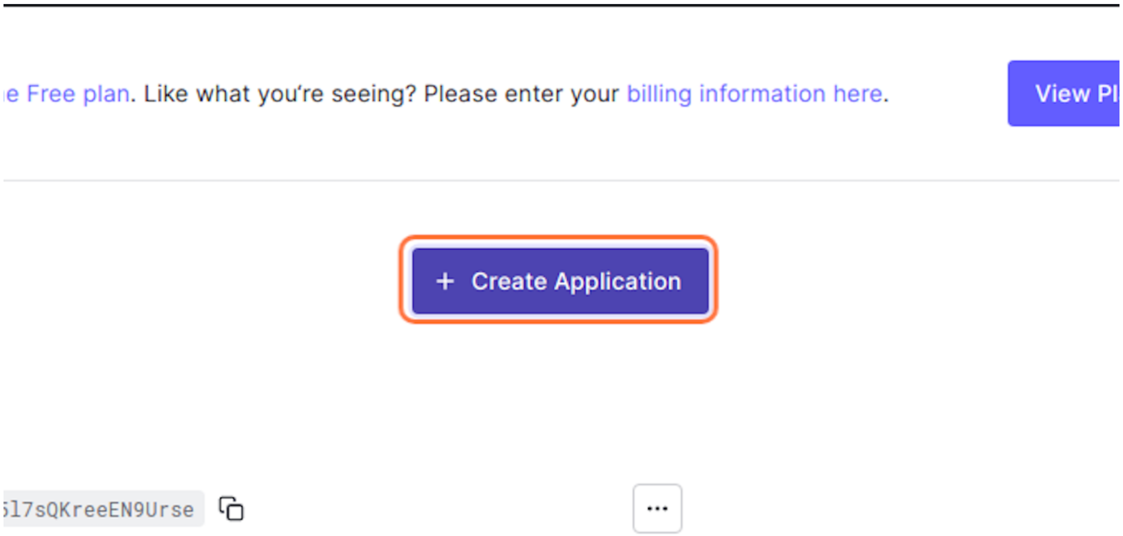
12. **Click Save Changes**



If enabled, Auth0 will prompt users to ente logging in. If disabled, you need to send th the appropriate Organization Login prompt

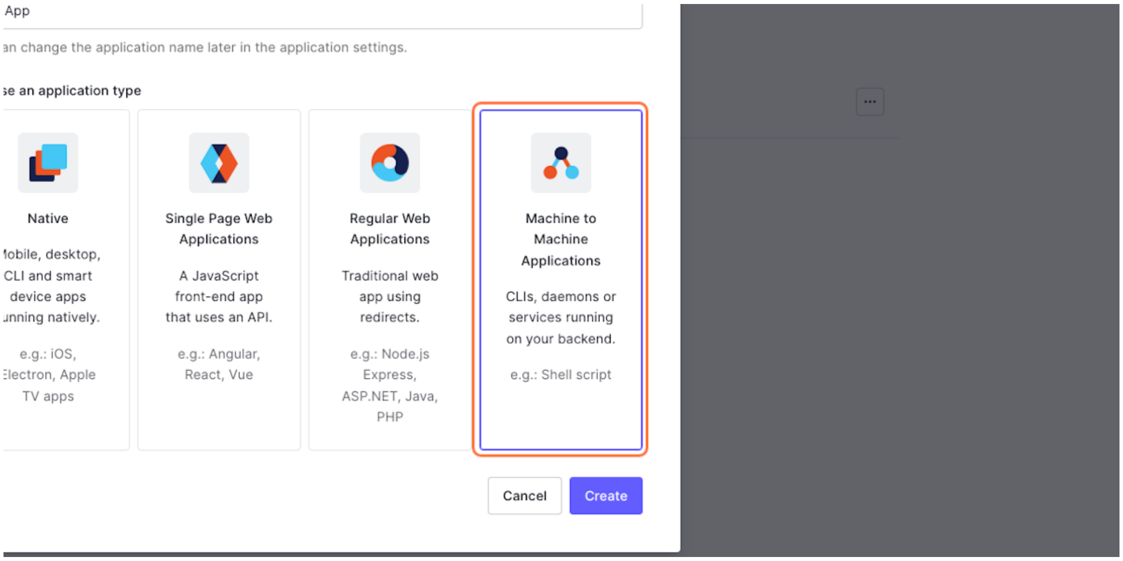
Create New Backend Application

1. **Click on Create Application**



e Free plan. Like what you're seeing? Please enter your billing information here.

2. **Select Machine to Machine Application**



App

an change the application name later in the application settings.

se an application type

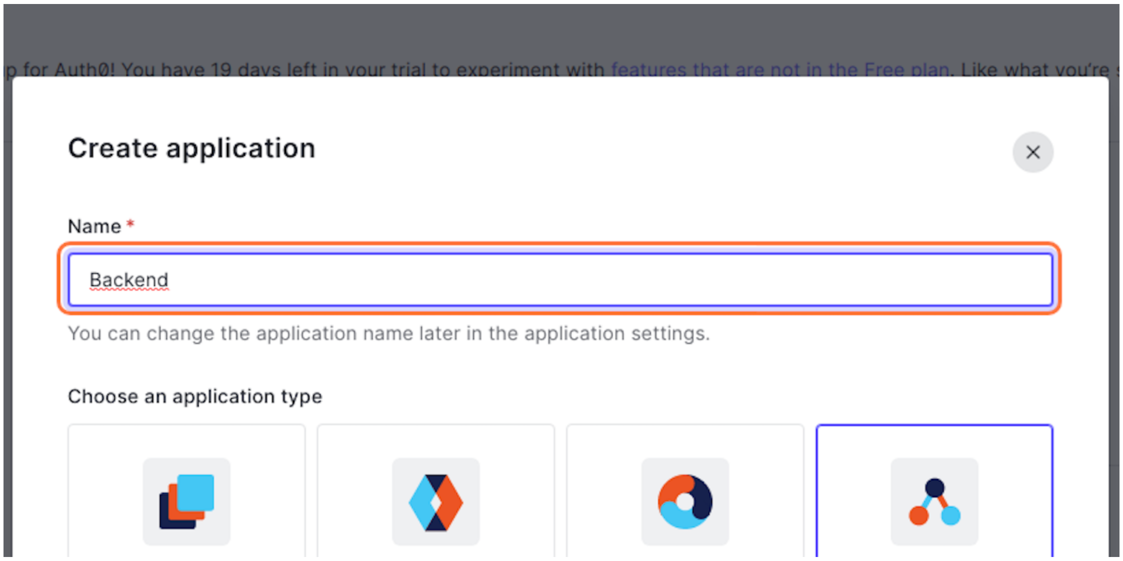
Native — 1obile, desktop, CLI and smart device apps unning natively. e.g.: iOS, Electron, Apple TV apps

Single Page Web Applications — A JavaScript front-end app that uses an API. e.g.: Angular, React, Vue

Regular Web Applications — Traditional web app using redirects. e.g.: Node.js Express, ASP.NET, Java, PHP

Machine to Machine Applications — CLIs, daemons or services running on your backend. e.g.: Shell script

3. **Rename the Application**



p for Auth0! You have 19 days left in your trial to experiment with features that are not in the Free plan. Like what you're s

Create application

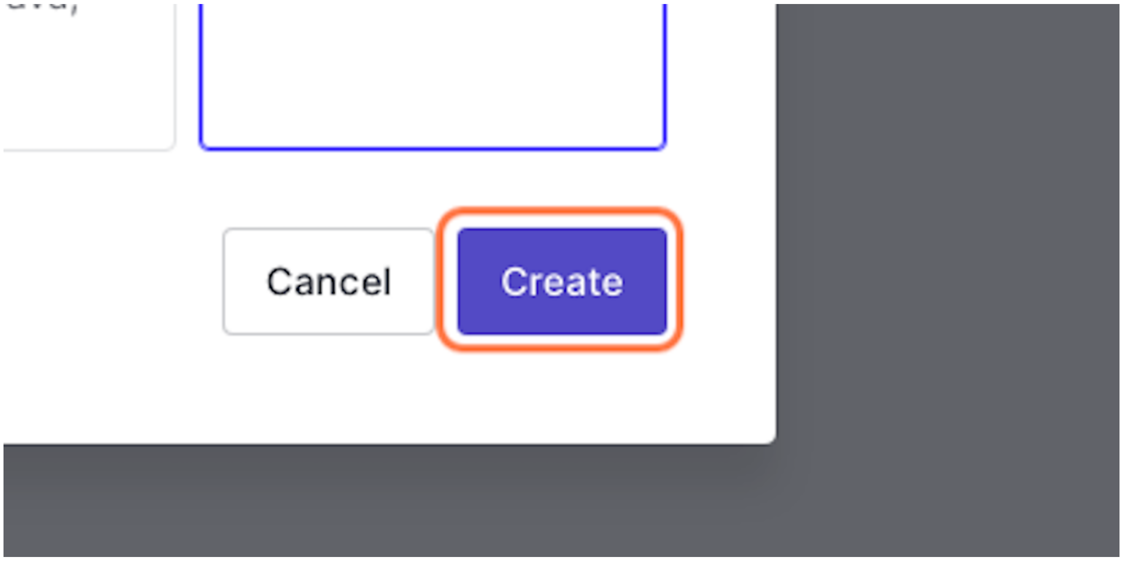
Name *

Backend

You can change the application name later in the application settings.

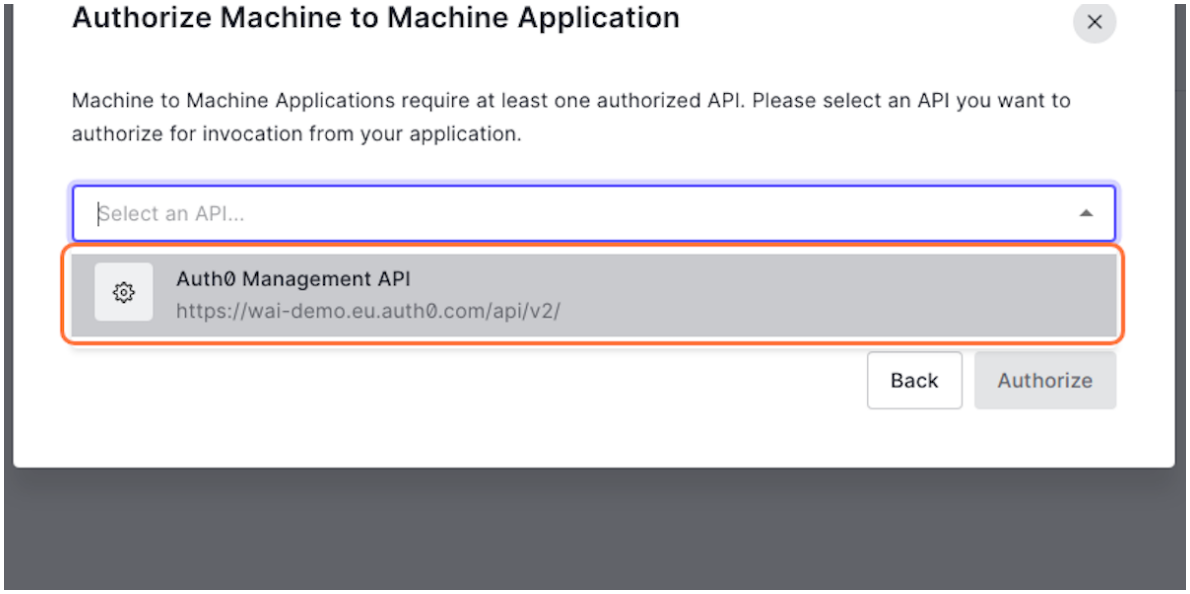
Choose an application type

4. **Click on Create**



5. **Click on option**

   Once you click create button, there will be a popup with dropdown selector to authorize this application. Please select **Auth0 Management API**



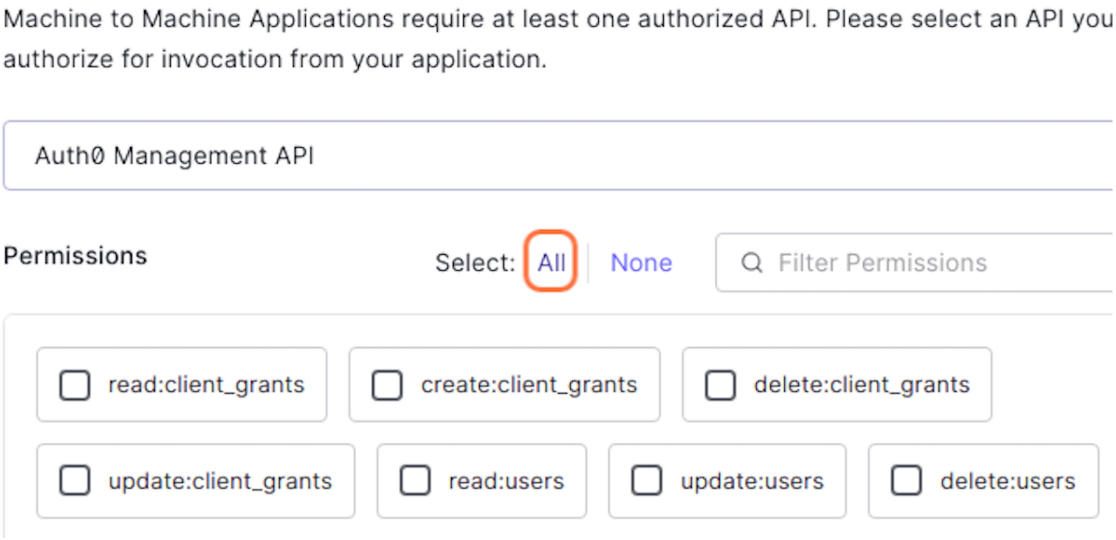
Authorize Machine to Machine Application

Machine to Machine Applications require at least one authorized API. Please select an API you want to authorize for invocation from your application.

Select an API...

Auth0 Management API
https://wai-demo.eu.auth0.com/api/v2/

6. Authorize All the Permissions



Machine to Machine Applications require at least one authorized API. Please select an API you authorize for invocation from your application.

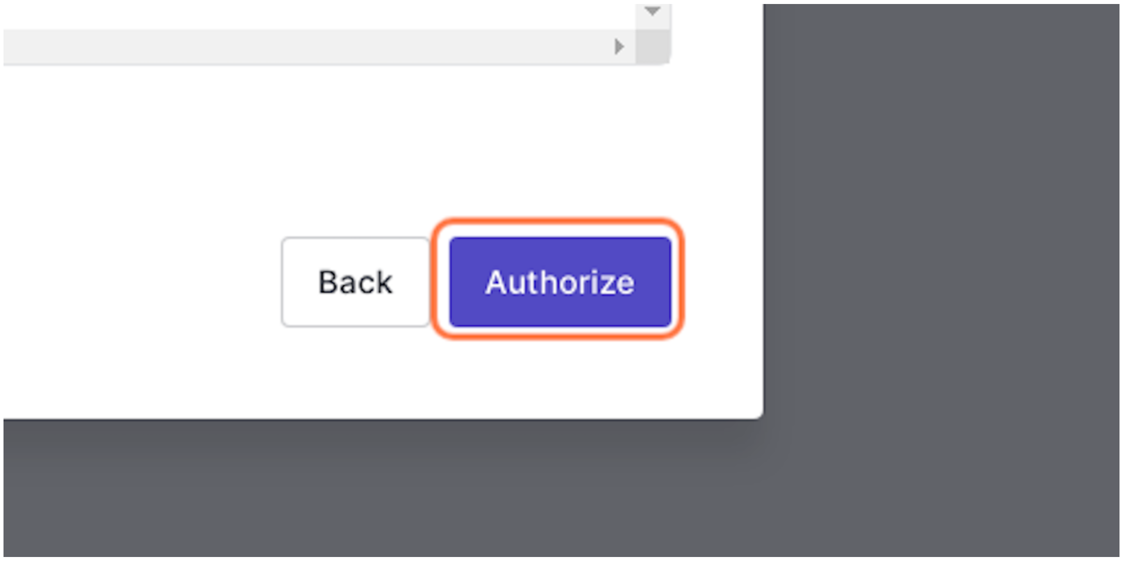
Auth0 Management API

Permissions

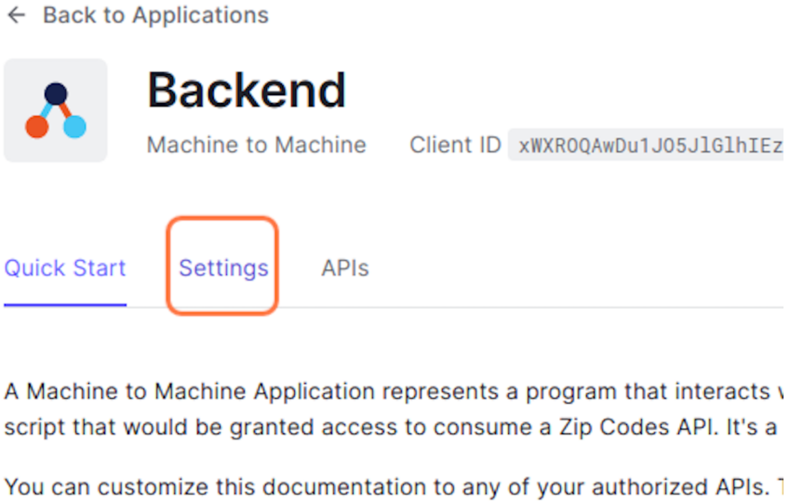
□ read:client_grants □ create:client_grants □ delete:client_grants

□ update:client_grants □ read:users □ update:users □ delete:users

7. Click on Authorize



8. Click on Settings



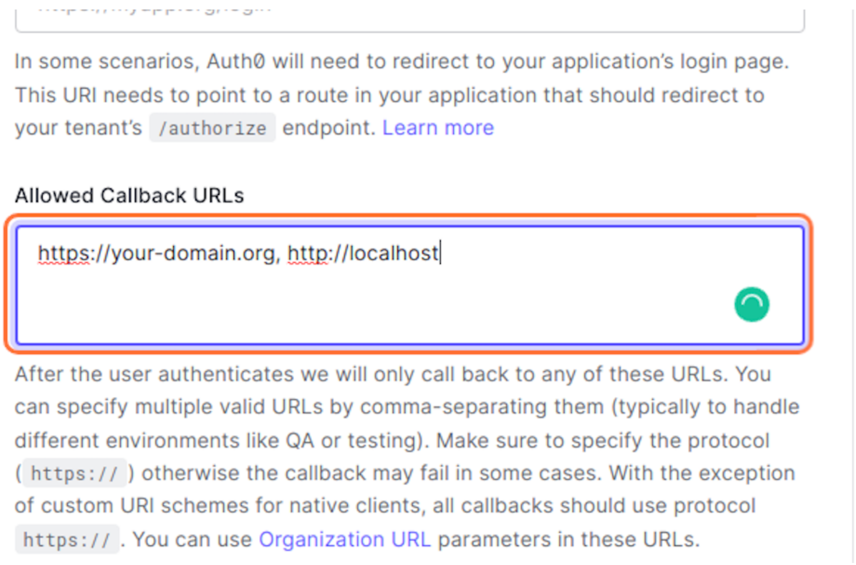
Backend

Machine to Machine Client ID xWXROQAwDu1J05J1GlhIEz

A Machine to Machine Application represents a program that interacts
script that would be granted access to consume a Zip Codes API. It's a

You can customize this documentation to any of your authorized APIs.

9. Change your-domain.com with your app-domain for all the field below

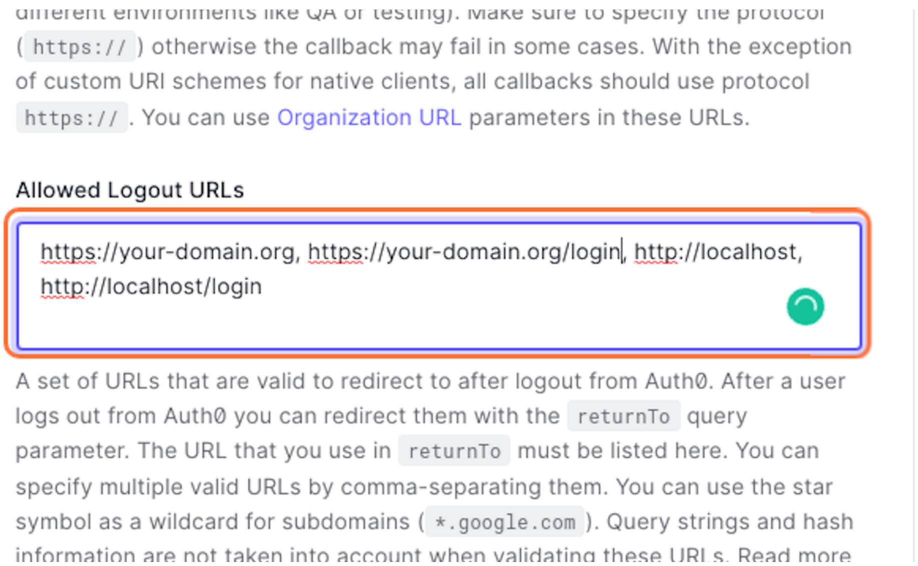


In some scenarios, Auth0 will need to redirect to your application's login page. This URI needs to point to a route in your application that should redirect to your tenant's `/authorize` endpoint. Learn more

Allowed Callback URLs

https://your-domain.org, http://localhost

After the user authenticates we will only call back to any of these URLs. You can specify multiple valid URLs by comma-separating them (typically to handle different environments like QA or testing). Make sure to specify the protocol (`https://`) otherwise the callback may fail in some cases. With the exception of custom URI schemes for native clients, all callbacks should use protocol `https://` . You can use Organization URL parameters in these URLs.

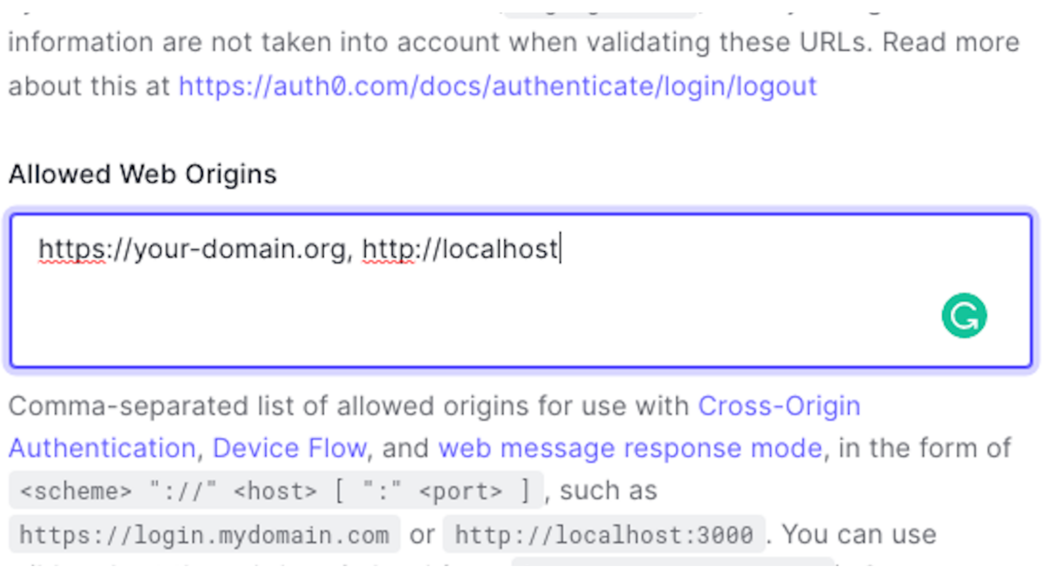


different environments like QA or testing). Make sure to specify the protocol (`https://`) otherwise the callback may fail in some cases. With the exception of custom URI schemes for native clients, all callbacks should use protocol `https://` . You can use Organization URL parameters in these URLs.

Allowed Logout URLs

https://your-domain.org, https://your-domain.org/login, http://localhost, http://localhost/login

A set of URLs that are valid to redirect to after logout from Auth0. After a user logs out from Auth0 you can redirect them with the `returnTo` query parameter. The URL that you use in `returnTo` must be listed here. You can specify multiple valid URLs by comma-separating them. You can use the star symbol as a wildcard for subdomains (`*.google.com`). Query strings and hash information are not taken into account when validating these URLs. Read more

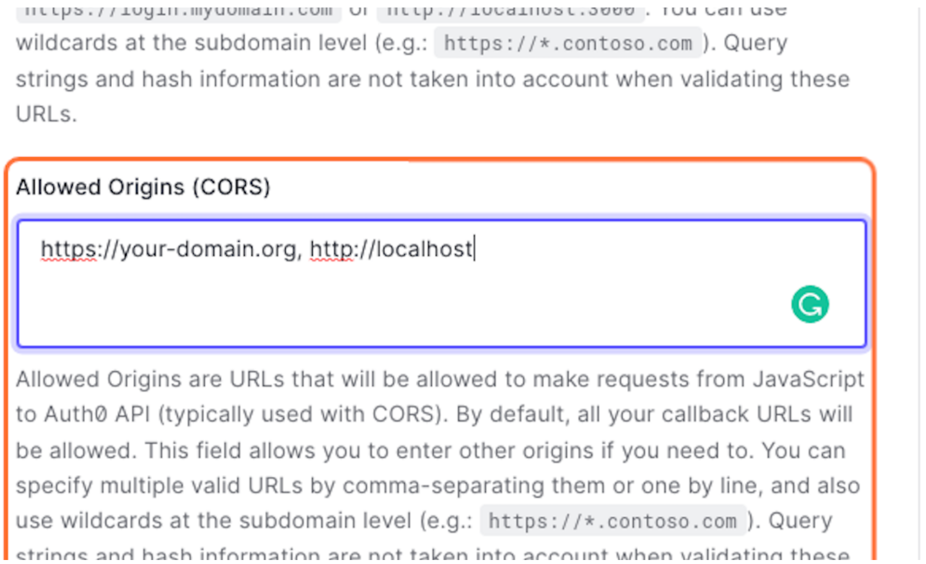


information are not taken into account when validating these URLs. Read more about this at https://auth0.com/docs/authenticate/login/logout

Allowed Web Origins

https://your-domain.org, http://localhost

Comma-separated list of allowed origins for use with Cross-Origin Authentication, Device Flow, and web message response mode, in the form of `<scheme> "://" <host> [ ":" <port> ]`, such as `https://login.mydomain.com` or `http://localhost:3000`. You can use



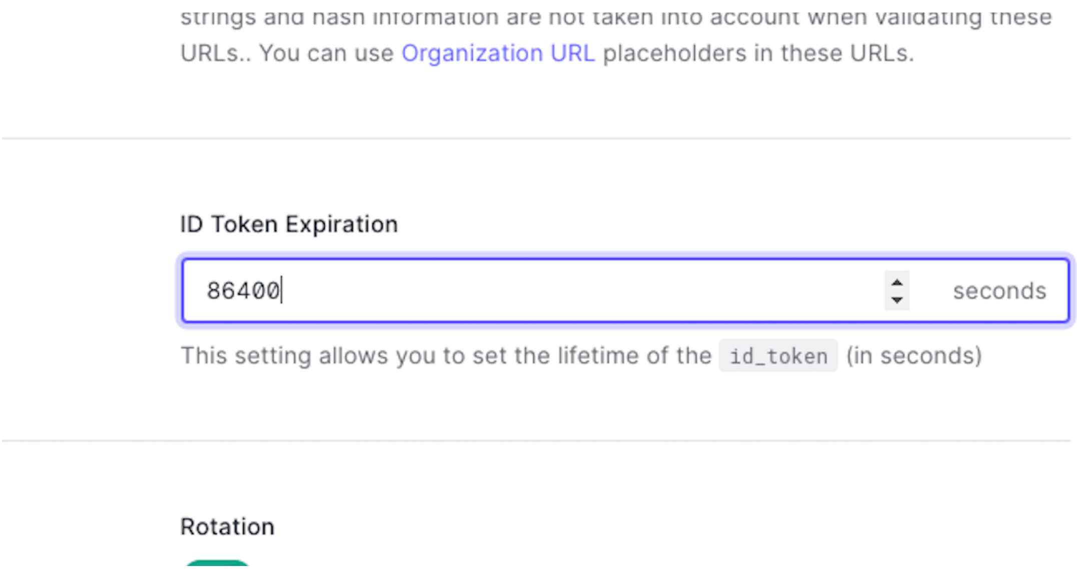
`https://login.mydomain.com` or `http://localhost:3000`. You can use
wildcards at the subdomain level (e.g.: `https://*.contoso.com`). Query strings and hash information are not taken into account when validating these URLs.

Allowed Origins (CORS)

https://your-domain.org, http://localhost

Allowed Origins are URLs that will be allowed to make requests from JavaScript to Auth0 API (typically used with CORS). By default, all your callback URLs will be allowed. This field allows you to enter other origins if you need to. You can specify multiple valid URLs by comma-separating them or one by line, and also use wildcards at the subdomain level (e.g.: `https://*.contoso.com`). Query strings and hash information are not taken into account when validating these

10. Modify the ID Token Expiration



strings and hash information are not taken into account when validating these URLs.. You can use Organization URL placeholders in these URLs.

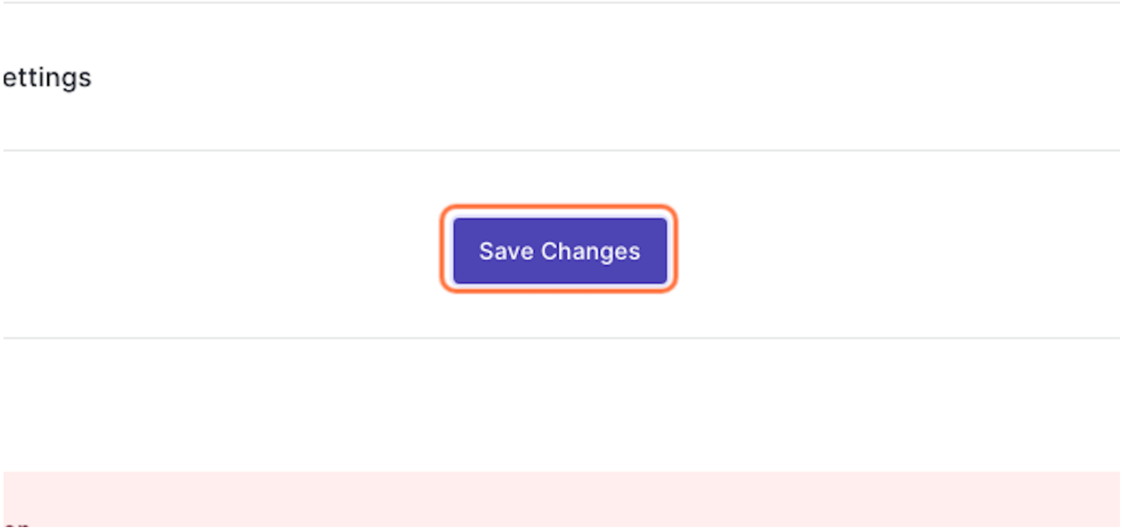
ID Token Expiration

86400 seconds

This setting allows you to set the lifetime of the `id_token` (in seconds)

Rotation

11. **Click Save Button**

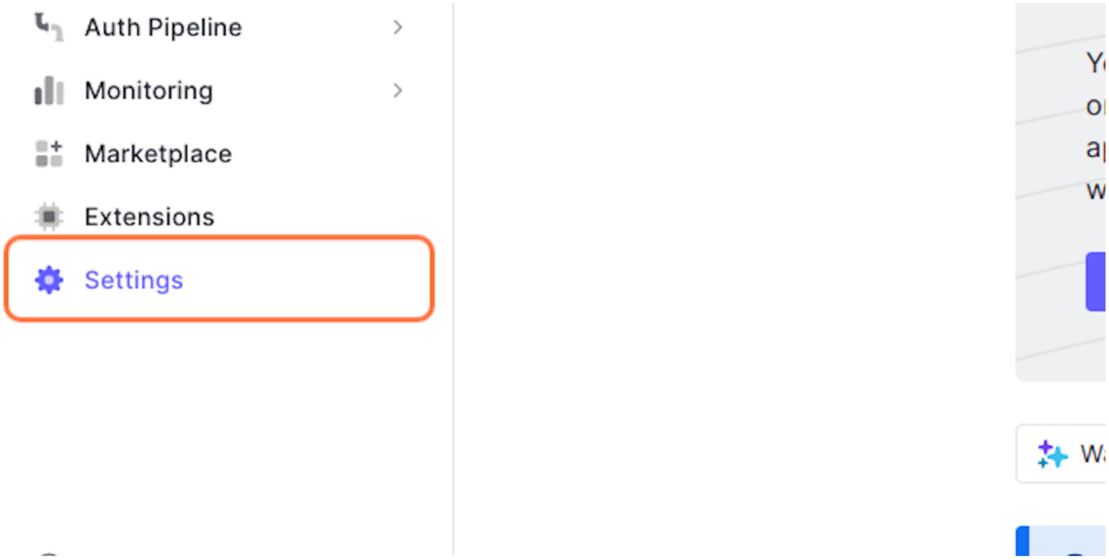


The Production Tentant

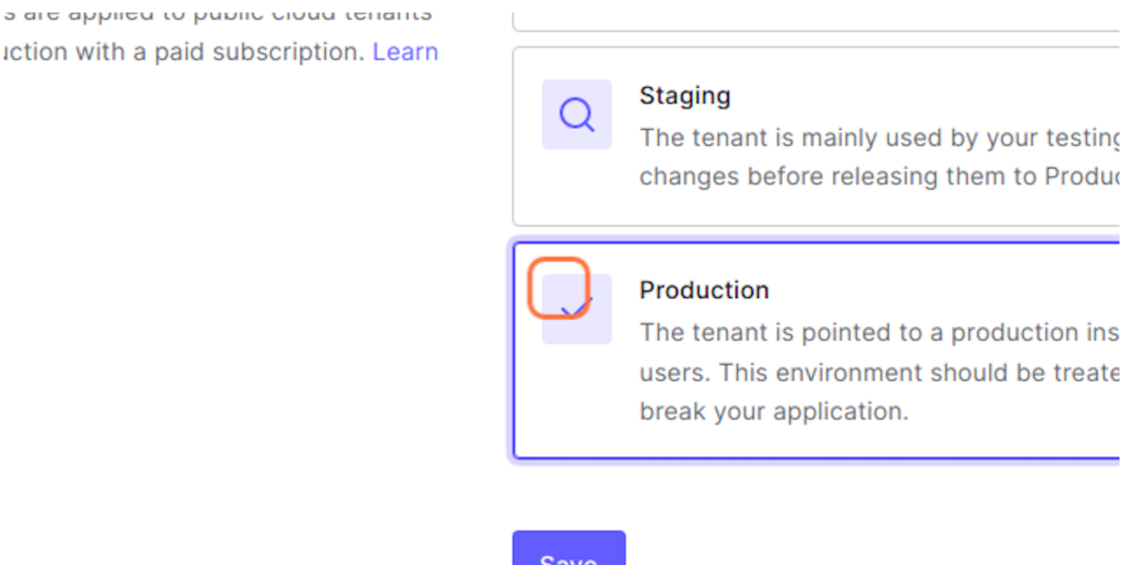
Tenants tagged as Production are granted higher rate limits than tenants tagged as Development or Staging. To ensure Auth0 recognizes your production tenant, be sure to set your production tenant with the **production** flag in the Support Center.

Note: Higher rate limits are applied to public cloud tenants tagged as Production with a paid subscription. See Auth0 Tenant Policy

1. **Click Gear Icons**



2. **Select Production**

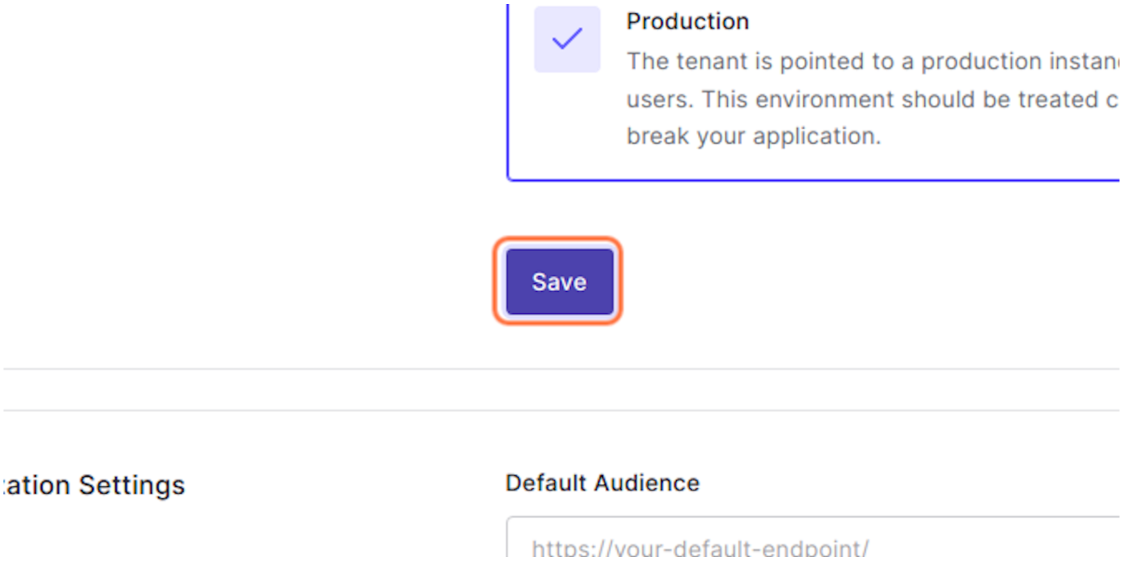


s are applied to public cloud tenants
ıction with a paid subscription. Learn

Staging
The tenant is mainly used by your testinç
changes before releasing them to Produc

Production
The tenant is pointed to a production ins
users. This environment should be treate
break your application.

3. **Click Save**



Production

The tenant is pointed to a production instan
users. This environment should be treated c
break your application.

ation Settings

Default Audience

https://your-default-endpoint/

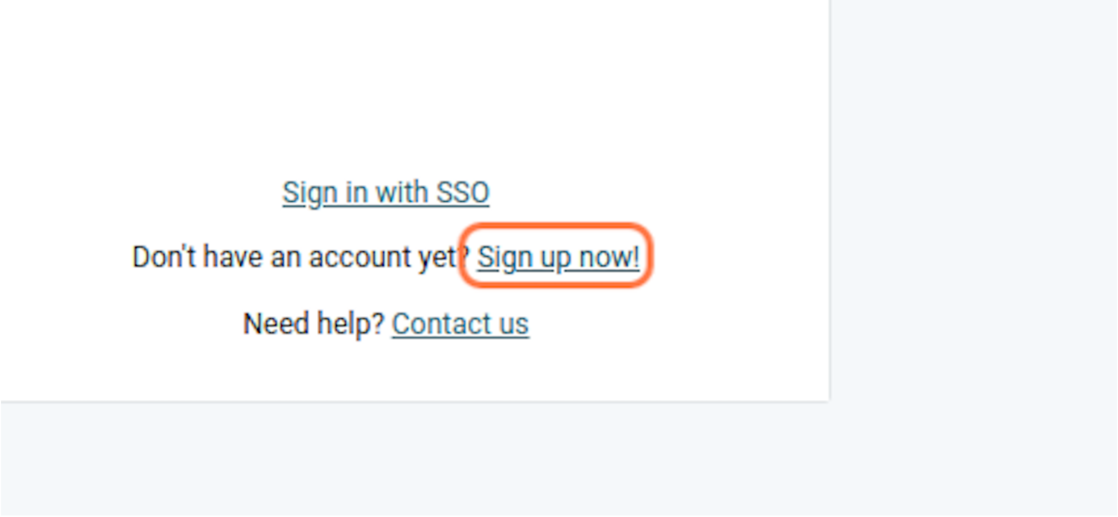
Back to Installation

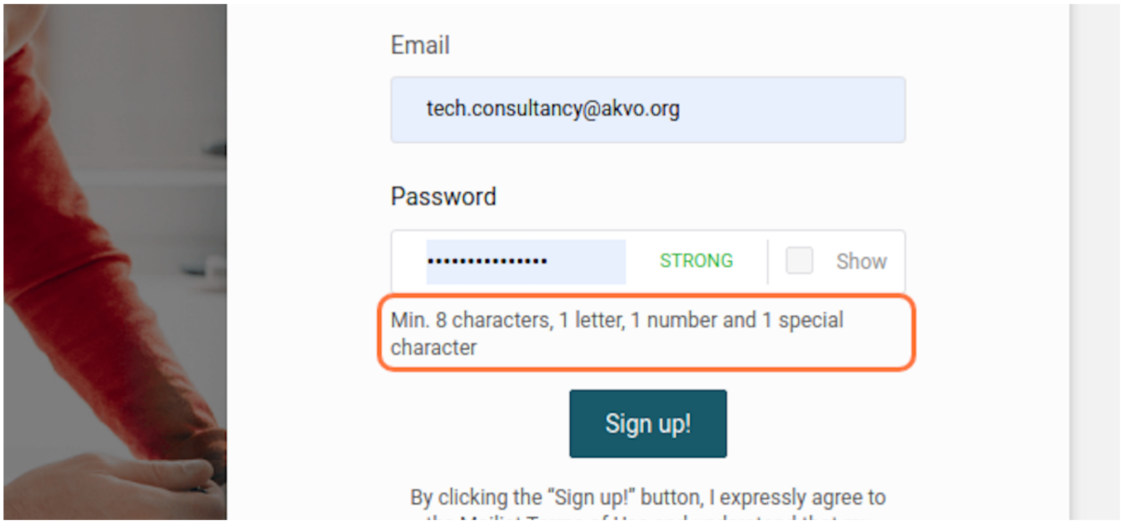
1.3.4 Mailjet Service

You need to have MAILJET account to manage the notification deliverability.

Create Mailjet Account

1. Click on Sign up now!





Email

tech.consultancy@akvo.org

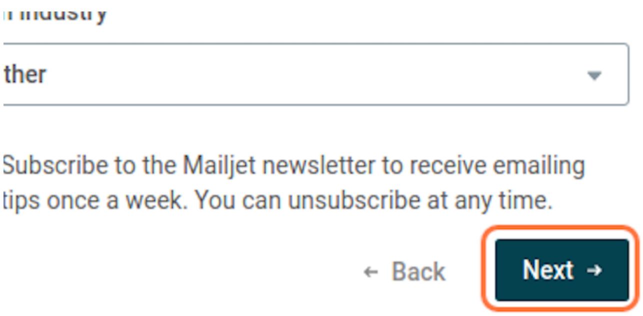
Password

•••••••••••••••• STRONG □ Show

Min. 8 characters, 1 letter, 1 number and 1 special character

By clicking the “Sign up!” button, I expressly agree to

2. Click on Next

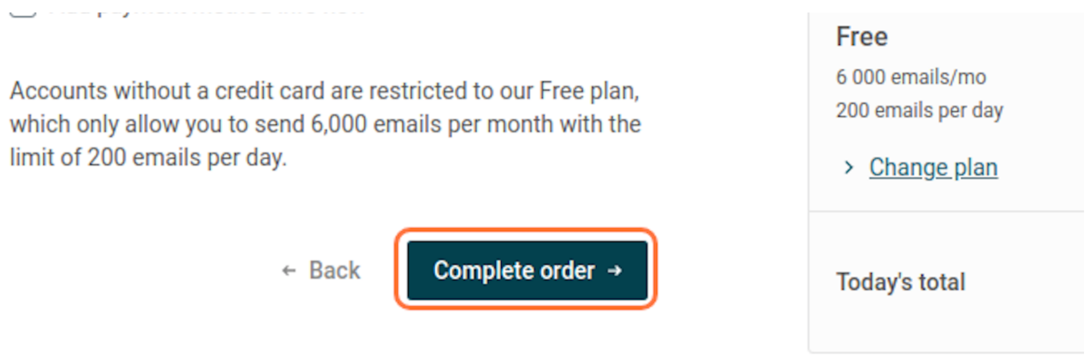


Industry

ther

Subscribe to the Mailjet newsletter to receive emailing tips once a week. You can unsubscribe at any time.

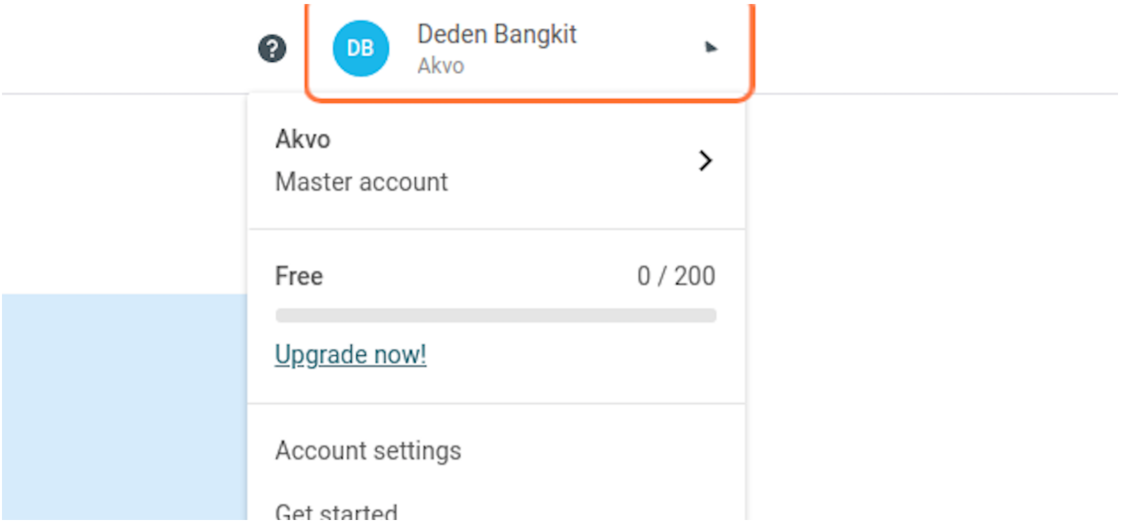
3. Click on Complete order then check your email for verification



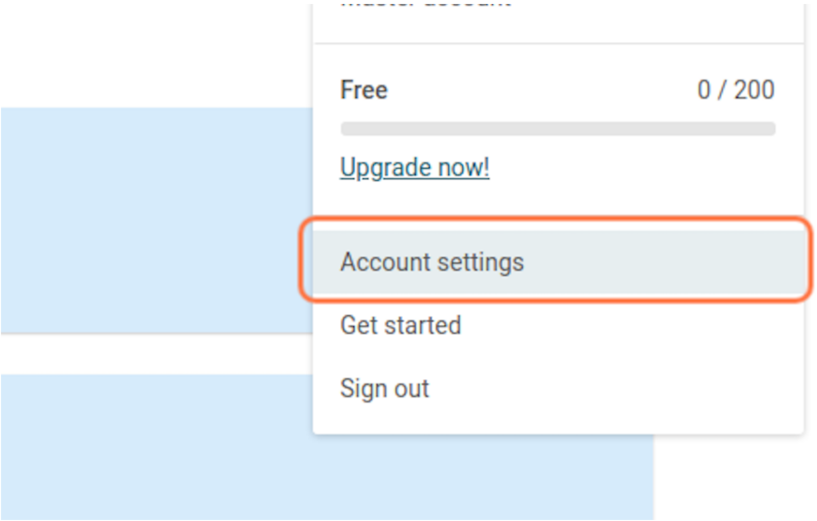
Accounts without a credit card are restricted to our Free plan, which only allow you to send 6,000 emails per month with the limit of 200 emails per day.

Generating API Key

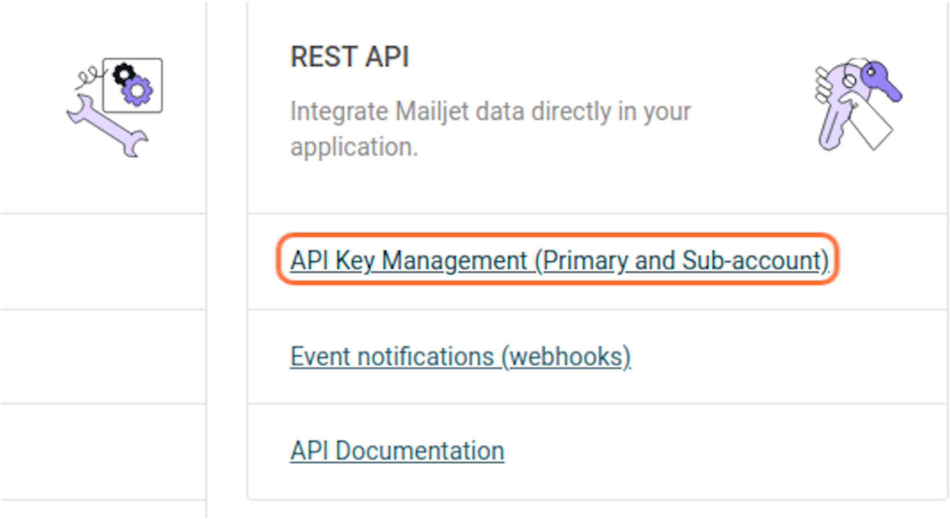
1. Click Account Dropdown



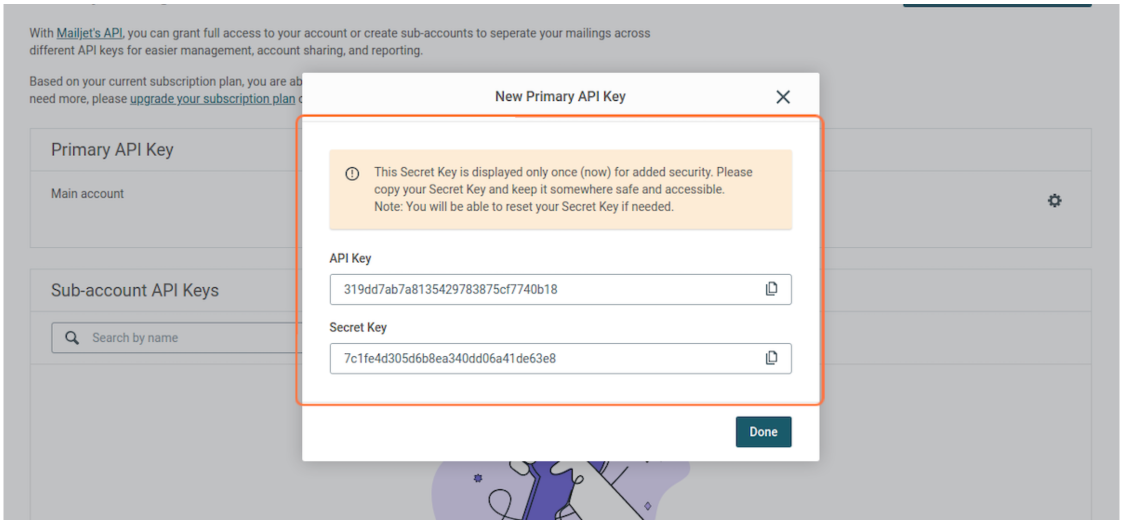
2. Click on Account settings



3. **Click on API Key Management**


REST API
Integrate Mailjet data directly in your application.
API Key Management (Primary and Sub-account)
Event notifications (webhooks)
API Documentation

4. **Click on Generate Secret Key**



With Mailjet's API, you can grant full access to your account or create sub-accounts to seperate your mailings across different API keys for easier management, account sharing, and reporting.

Based on your current subscription plan, you are ab
need more, please upgrade your subscription plan c

Primary API Key

Main account

Sub-account API Keys

New Primary API Key

This Secret Key is displayed only once (now) for added security. Please copy your Secret Key and keep it somewhere safe and accessible.
Note: You will be able to reset your Secret Key if needed.

API Key: 319dd7ab7a8135429783875cf7740b18

Secret Key: 7c1fe4d305d6b8ea340dd06a41de63e8

1.4 Installation

1.4.1 Clone the Repository

```
git clone https://github.com/akvo/wai-sdg-portal.git
```

1.4.2 Environment Variable Setup

Install text editor to be able to edit *.env* file

```
sudo apt install nano
```

or

```
sudo apt install vim
```

Go to the repository directory, then edit the environment

```
cd wai-sdg-portal/deploy
vim .env
```

Example Environemnt:

```
POSTGRES_PASSWORD=postgres
WAI_DB_USER=yourname
WAI_DB_PASSWORD=sUpeRsTr0ngPa**word
INSTANCE_NAME=wai-demo
AUTH0_DOMAIN=your-domain.eu.auth0.com
AUTH0_CLIENT_ID=acad34xxxxxxxx
AUTH0_SECRET=938axxxxxxxxxxx
AUTH0_AUDIENCE=cdary8xxxxxxxx
```

(continues on next page)

(continued from previous page)

```
AUTH0_SPA_DOMAIN=5a2axxxxxxxxxxx
AUTH0_SPA_CLIENT_ID=b821y8xxxxxxxx
STORAGE_LOCATION=/data/storage
MAILJET_SECRET=093asbalxxxxxxxx
MAILJET_APIKEY=9acadlkbxxxxxxxx
WEBDOMAIN=https://your-domain.com
```

Note:

- Use **Domain** and **Client ID** field from your **Auth0 SPA application** for `AUTH_SPA_DOMAIN` and `AUTH_SPA_CLIENT_ID`
- Use **Domain**, **Secret** and **Client ID** field from your **Auth0 Backend application** for `AUTH_DOMAIN`, `AUTH_SECRET` and `AUTH_CLIENT_ID`.
- For `AUTH0_AUDIENCE`, Go to your **Auth0 backend application**, click **APIs** Tab, expand **Auth0 Management API**. Use the **Grant ID** field.

1.4.3 Run the Application

```
./install.sh
```

1.4.4 Post-Installation

Once the app is started, we need to populate the database with the initial data set. The initial dataset are:

- 1st Super Admin
- 1st Organisation
- Administration Levels Data

Run the database seeder:

```
docker compose exec backend ./seed.sh youremail@akvo.org "Your Full Name" "Your Organisation"
```

Example:

```
docker compose exec backend ./seed.sh youremail@akvo.org "Your Name" Akvo
```

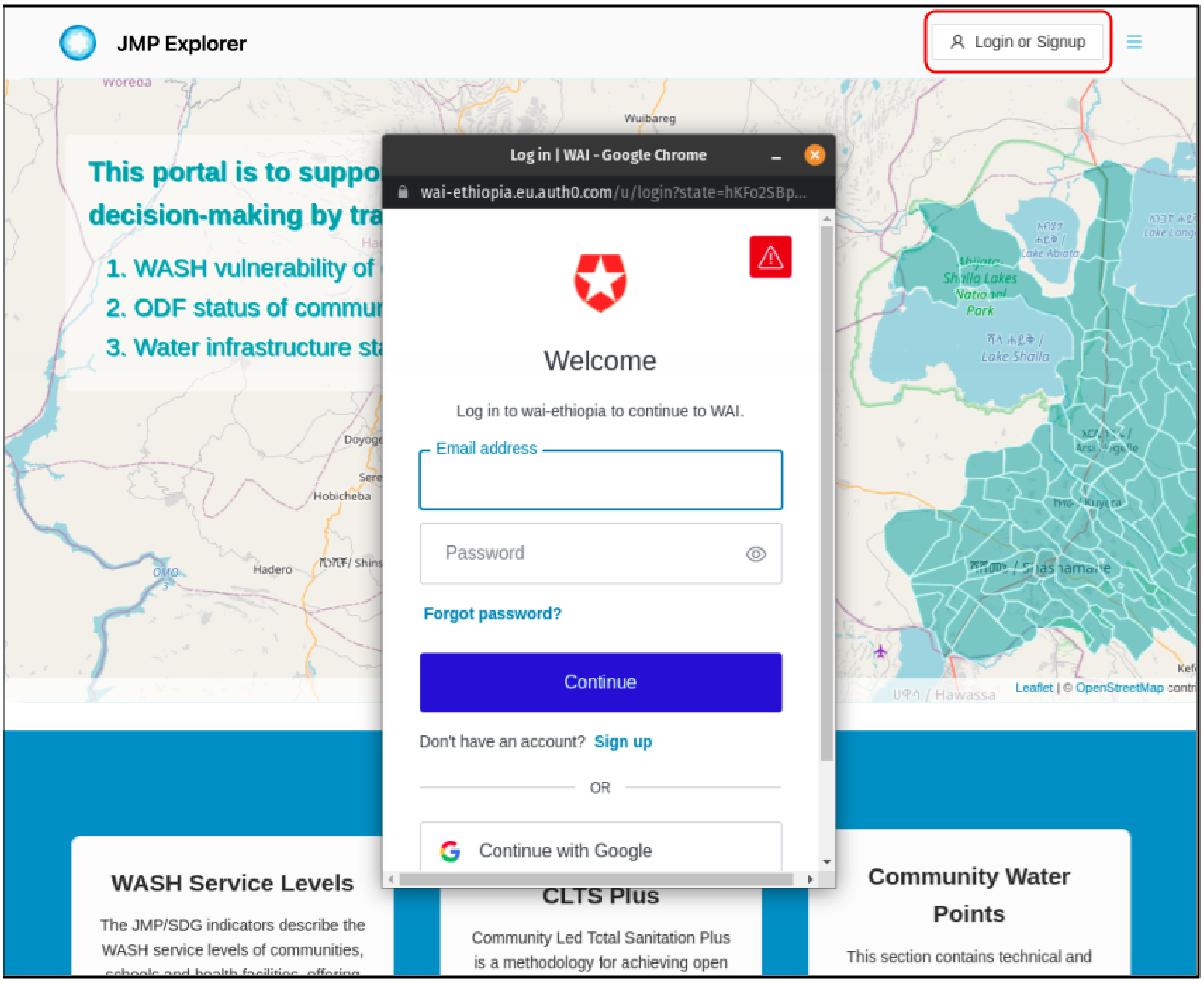
CHAPTER

TWO

GETTING STARTED

2.1 Log in

Before, you start looking around and going into deep, please logging in by clicking on the **Login or Signup**.



This portal is to suppo
decision-making by tra

1. WASH vulnerability of
2. ODF status of commun
3. Water infrastructure sta

Log in to wai-ethiopia to continue to WAI.

Don't have an account? Sign up

The JMP/SDG indicators describe the WASH service levels of communities, schools and health facilities, offering

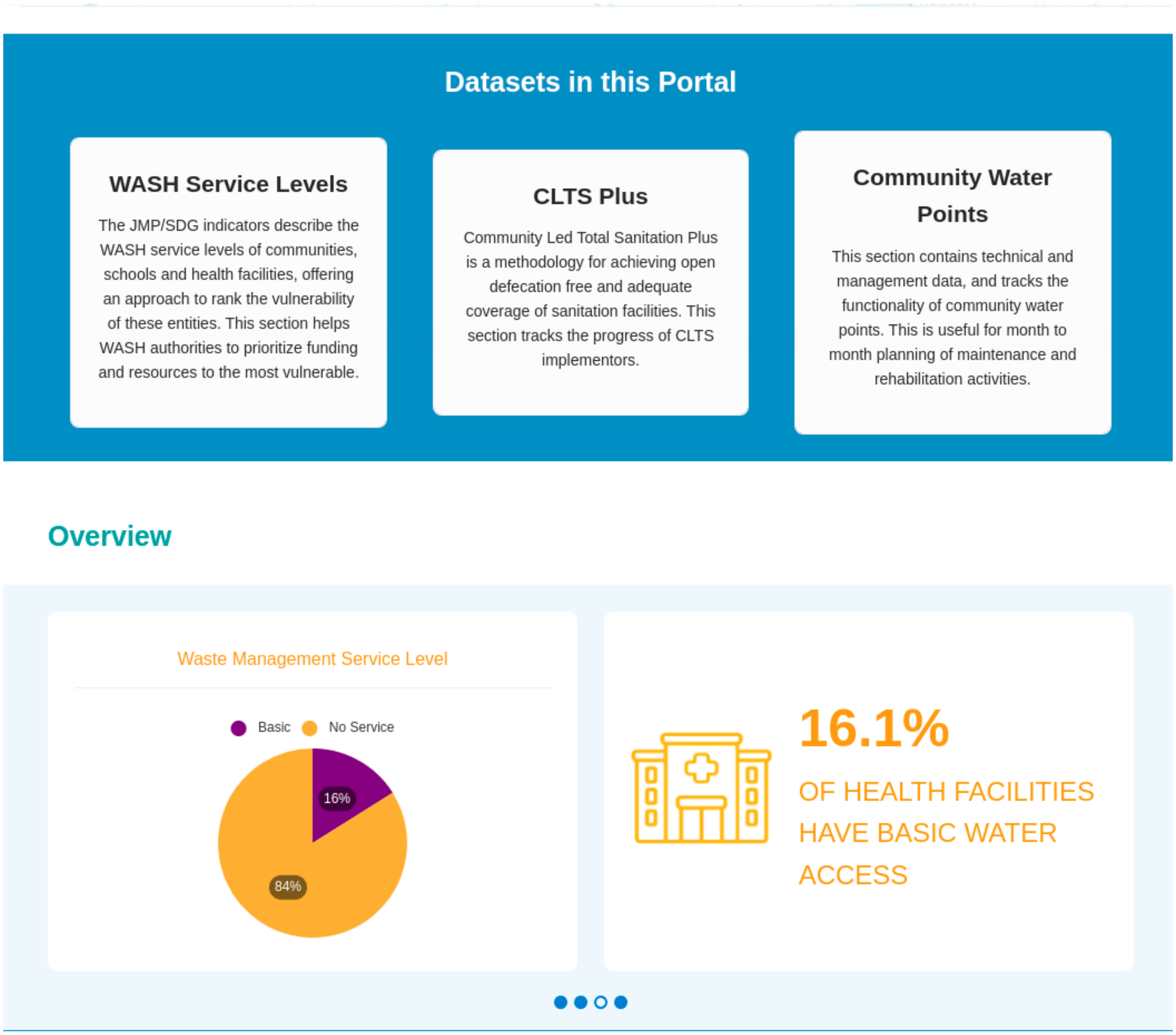
Community Led Total Sanitation Plus is a methodology for achieving open

This section contains technical and

CHAPTER THREE

DATASETS

Take a quick look at the datasets.



Datasets in this Portal

WASH Service Levels

The JMP/SDG indicators describe the WASH service levels of communities, schools and health facilities, offering an approach to rank the vulnerability of these entities. This section helps WASH authorities to prioritize funding and resources to the most vulnerable.

CLTS Plus

Community Led Total Sanitation Plus is a methodology for achieving open defecation free and adequate coverage of sanitation facilities. This section tracks the progress of CLTS implementors.

Community Water Points

This section contains technical and management data, and tracks the functionality of community water points. This is useful for month to month planning of maintenance and rehabilitation activities.

Overview

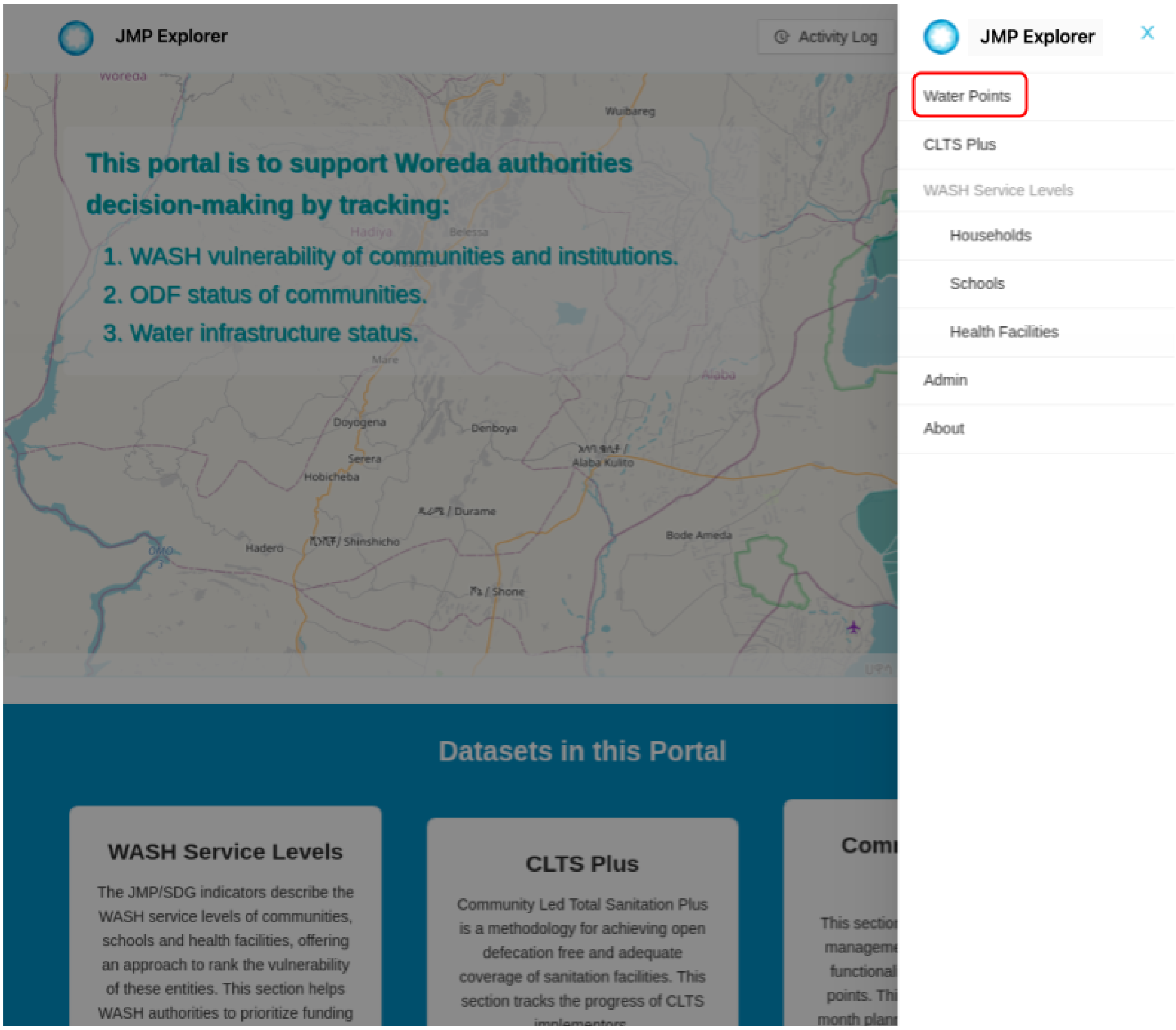
Waste Management Service Level

16.1%

OF HEALTH FACILITIES HAVE BASIC WATER ACCESS

3.1 List of datasets

A list of datasets is shown once you click on the **Menu** button



This portal is to support Woreda authorities decision-making by tracking:

1. WASH vulnerability of communities and institutions.
2. ODF status of communities.
3. Water infrastructure status.

Datasets in this Portal

WASH Service Levels

The JMP/SDG indicators describe the WASH service levels of communities, schools and health facilities, offering an approach to rank the vulnerability of these entities. This section helps WASH authorities to prioritize funding

CLTS Plus

Community Led Total Sanitation Plus is a methodology for achieving open defecation free and adequate coverage of sanitation facilities. This section tracks the progress of CLTS implementors

Comı

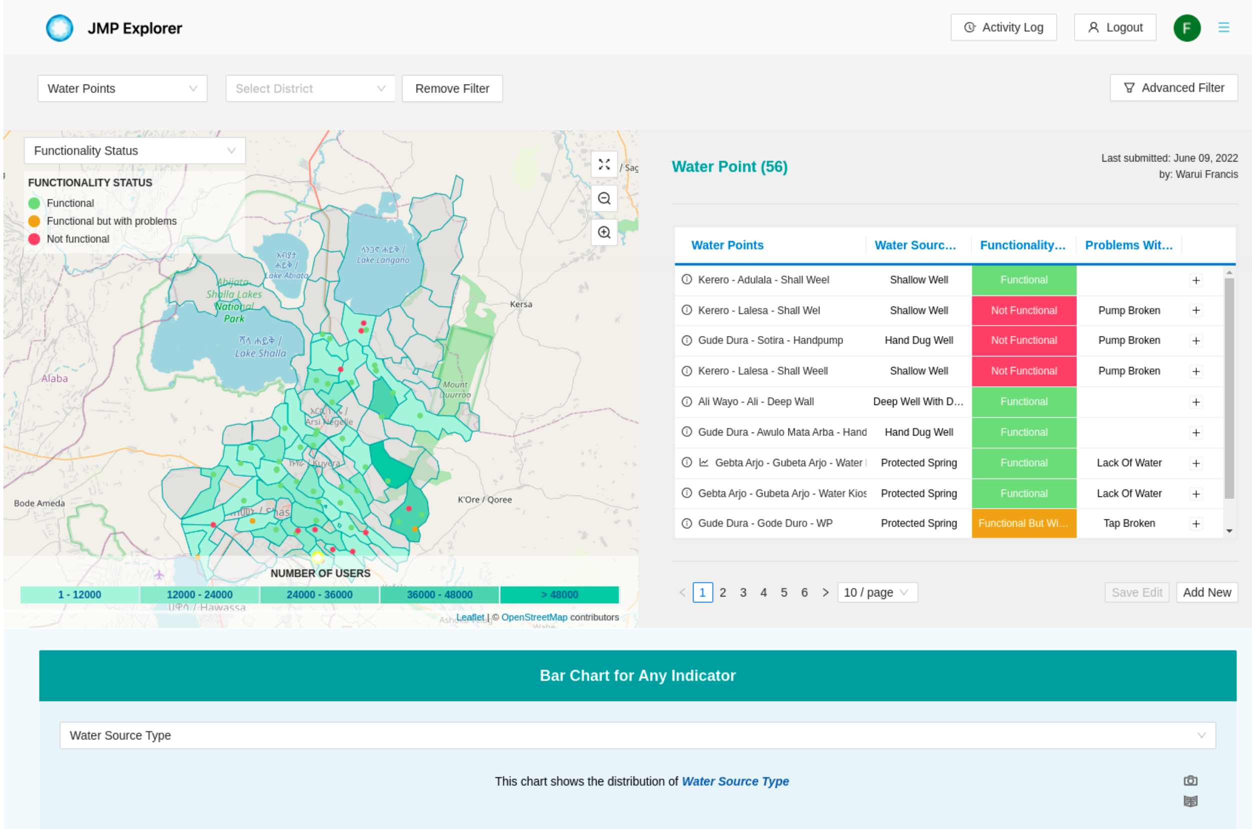
This sectio manageme functional points. Thi month plan

CHAPTER

FOUR

DATA

Each data has multiple questions and question groups.



Water Point (56)

Last submitted: June 09, 2022
by: Warui Francis

Water Points	Water Sourc...	Functionality...	Problems Wit...	
Kerero - Adulala - Shall Weel	Shallow Well	Functional		+
Kerero - Lalesa - Shall Wel	Shallow Well	Not Functional	Pump Broken	+
Gude Dura - Sotira - Handpump	Hand Dug Well	Not Functional	Pump Broken	+
Kerero - Lalesa - Shall Weell	Shallow Well	Not Functional	Pump Broken	+
Ali Wayo - Ali - Deep Wall	Deep Well With D...	Functional		+
Gude Dura - Awulo Mata Arba - Hand	Hand Dug Well	Functional		+
Gebta Arjo - Gubeta Arjo - Water	Protected Spring	Functional	Lack Of Water	+
Gebta Arjo - Gubeta Arjo - Water Kios	Protected Spring	Functional	Lack Of Water	+
Gude Dura - Gode Duro - WP	Protected Spring	Functional But Wi...	Tap Broken	+

< 1 2 3 4 5 6 > 10 / page   Save Edit   Add New

Bar Chart for Any Indicator

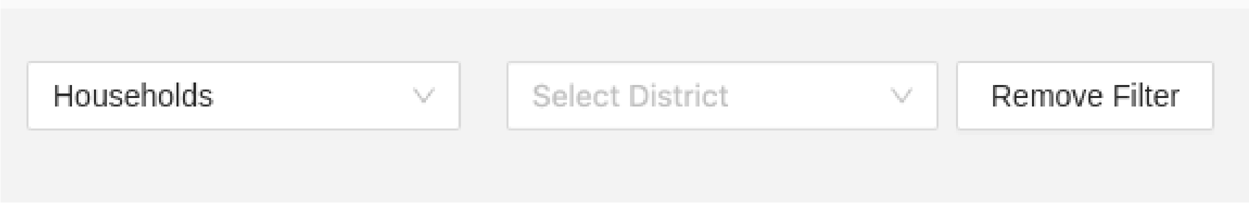
Water Source Type

This chart shows the distribution of ***Water Source Type***

4.1 Search

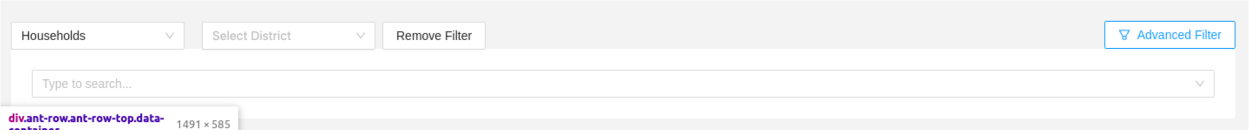
4.1.1 Selecting

- Filtering a data point by selecting name, district, sub-district and village
- You reset your search by clicking the **REMOVE FILTER** button.



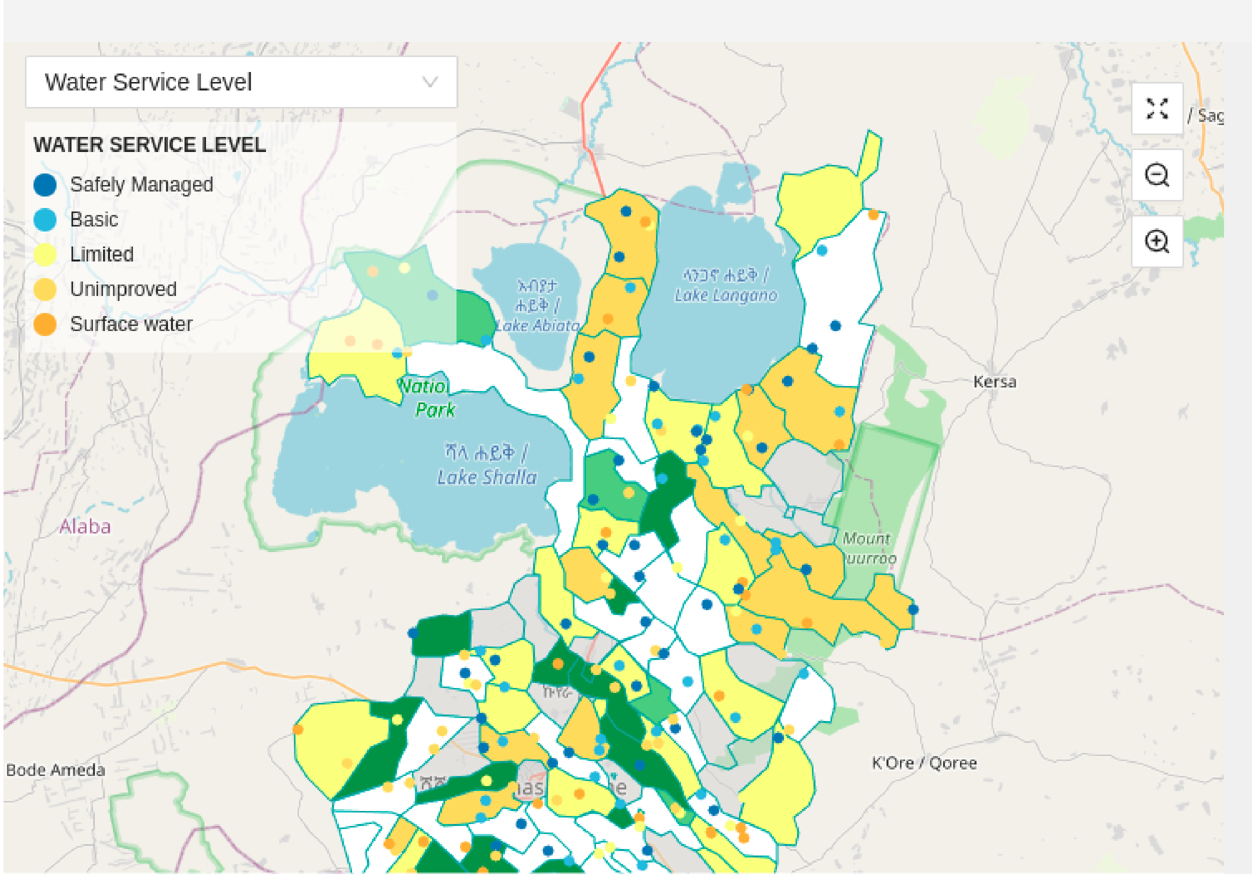
4.1.2 Advanced search

If you wish to search in more advanced way, an **Advanced filter** button is clickable that shows an input select.



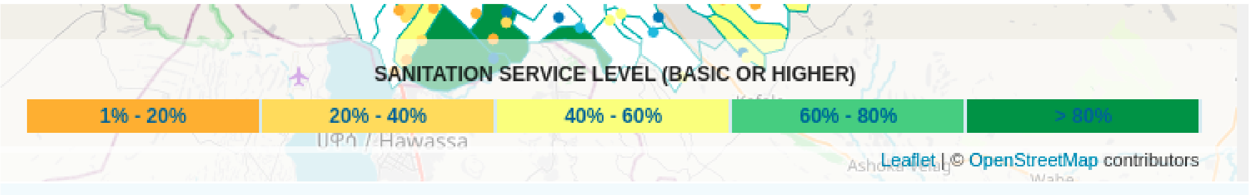
4.2 Maps

- A map is on the page which shows all places, district, sub-district and village from where the data is collected.
- This map can be zoomed out or in



4.2.1 Maps Legend

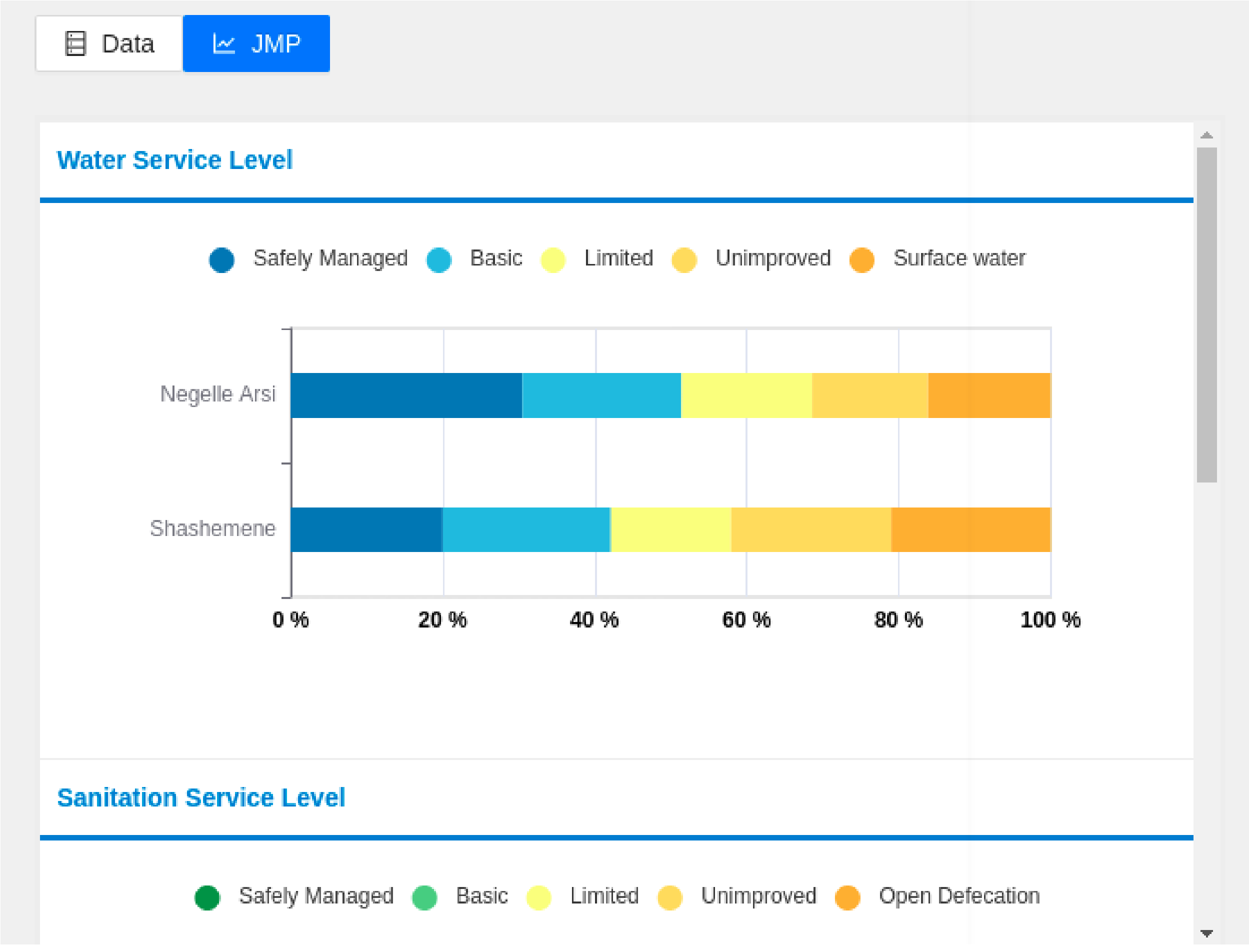
A legend that displays the level of data services is shown on the page under the map.


SANITATION SERVICE LEVEL (BASIC OR HIGHER)
1% - 20%
20% - 40%
40% - 60%
60% - 80%
> 80%
ሀዋሳ / Hawassa
Leaflet | © OpenStreetMap contributors

4.3 Chart data

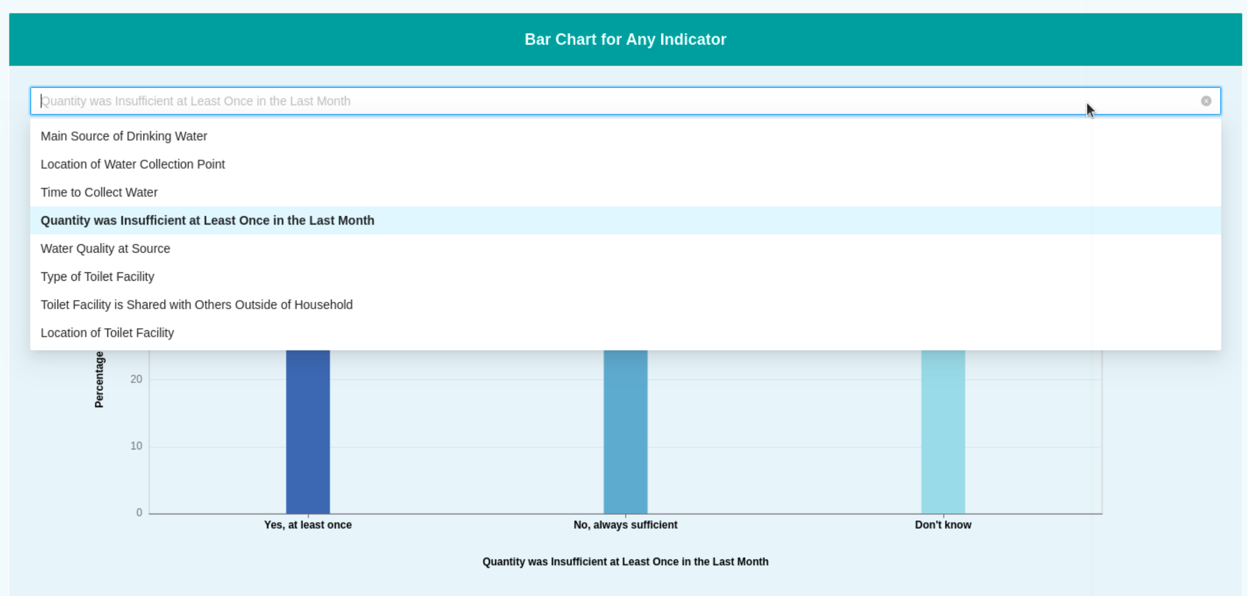
4.3.1 JMP

A chart is also on the page once the **JMP tab** is active



4.3.2 Bar

- A disruption of a question is shown on a **Bar char**. To see this, you have to select a question.
- This bar chart can also be saved as an image by clickin on the **SAVE IMAGE** icon.
- If you would love to see the question value on a table, there is table icon that can be clicked to do so.


Bar Chart for Any Indicator
Quantity was Insufficient at Least Once in the Last Month
Main Source of Drinking Water
Location of Water Collection Point
Time to Collect Water
Quantity was Insufficient at Least Once in the Last Month
Water Quality at Source
Type of Toilet Facility
Toilet Facility is Shared with Others Outside of Household
Location of Toilet Facility
Percentage
20
10
0
Yes, at least once
No, always sufficient
Don't know
Quantity was Insufficient at Least Once in the Last Month

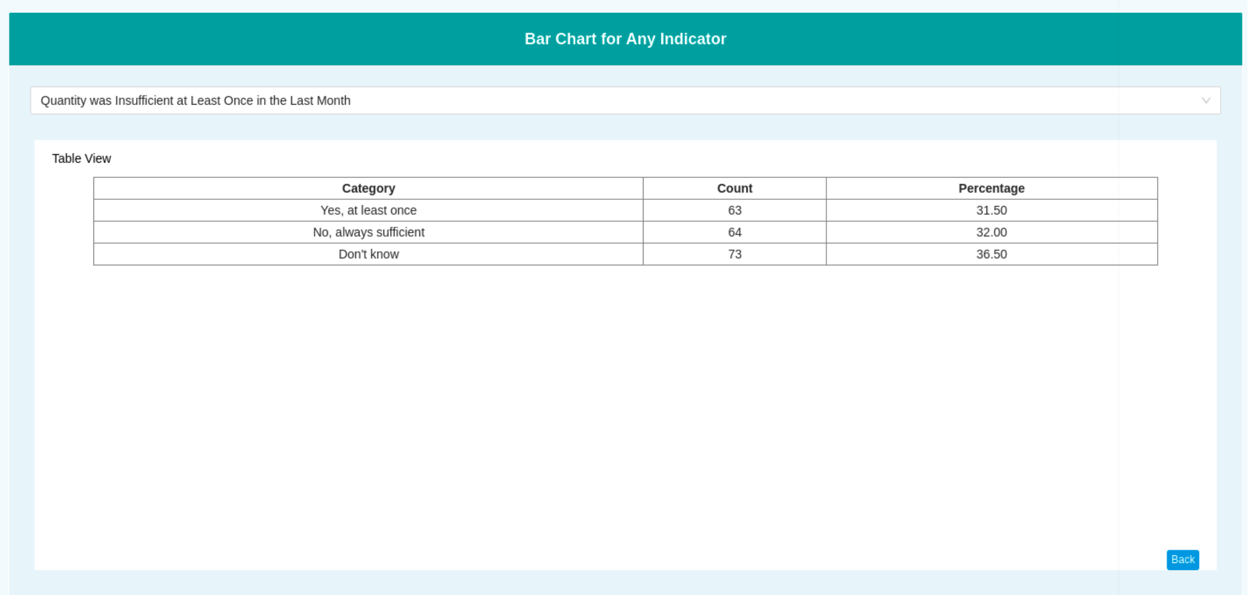

Bar Chart for Any Indicator
Quantity was Insufficient at Least Once in the Last Month
Back

Table View

Category	Count	Percentage
Yes, at least once	63	31.50
No, always sufficient	64	32.00
Don't know	73	36.50

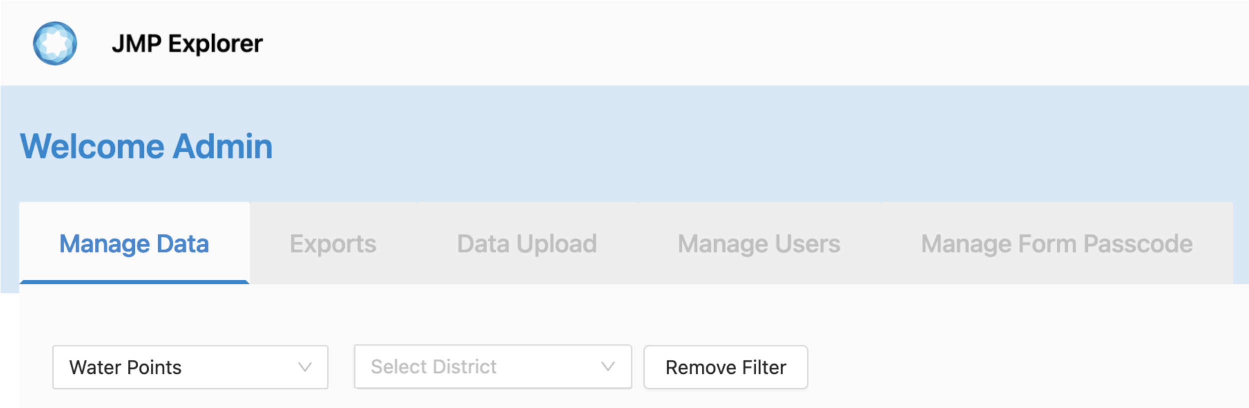
CHAPTER FIVE

ADMIN PAGE

5.1 Manage data

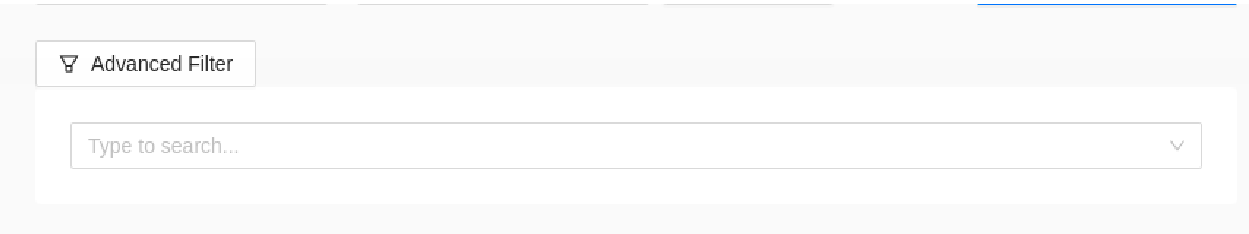
5.1.1 Filter

Searching data by - data point - District, sub-district and village - You can also reset your filter search.

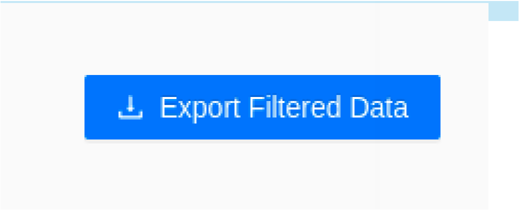


5.1.2 Advanced filter

If you wish to search in a more advanced way, you click on the **ADVANCED FILTER** button.

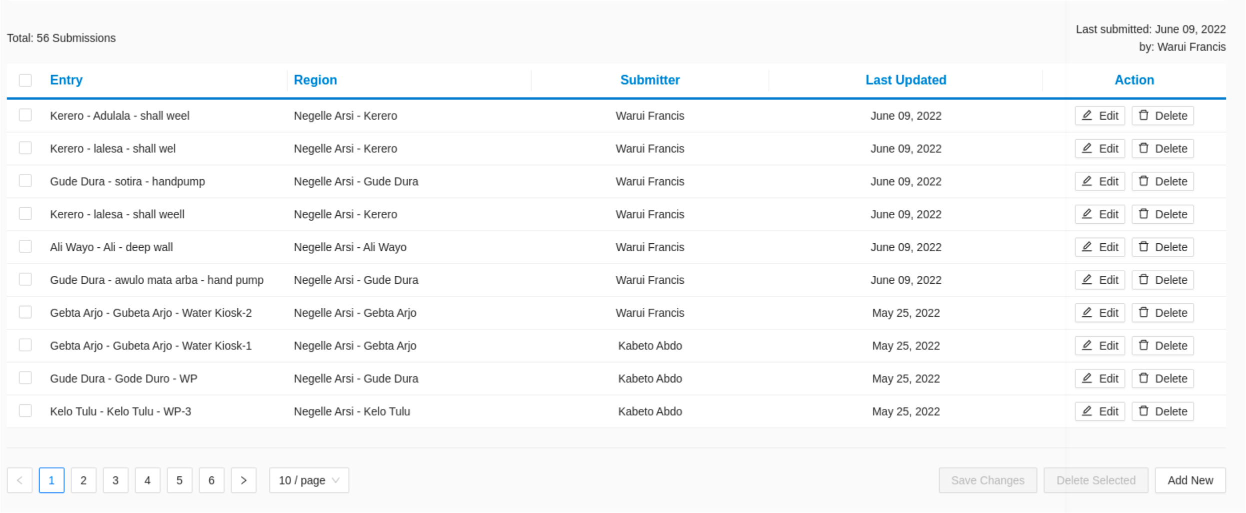


5.1.3 Export filtered data



5.1.4 List of data

A list of data is on the page as a table which has - data name - Region - Submitter - Last updated - Action



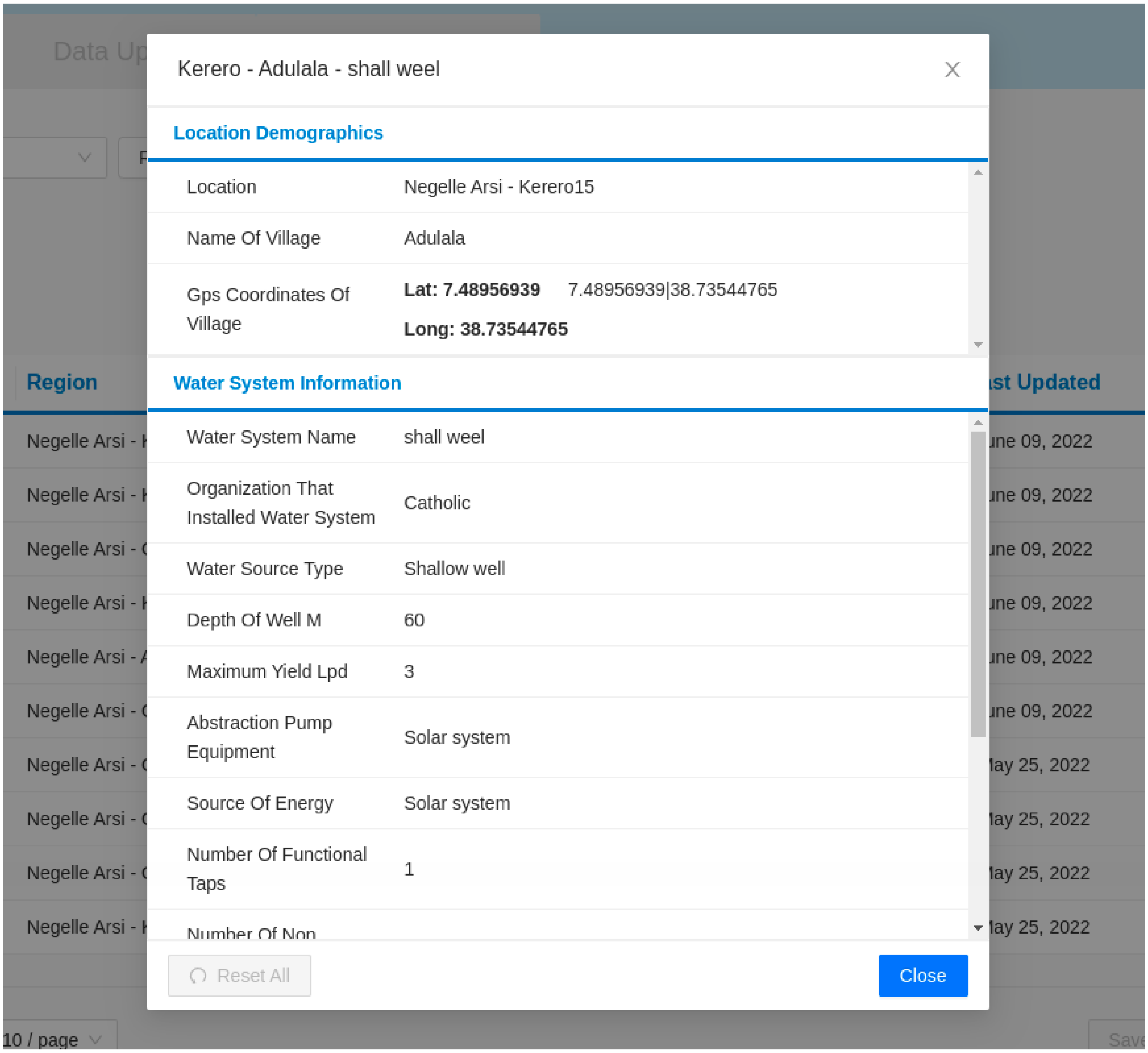
Total: 56 Submissions

Last submitted: June 09, 2022
by: Warui Francis

□	Entry	Region	Submitter	Last Updated	Action
□	Kerero - Adulala - shall weel	Negelle Arsi - Kerero	Warui Francis	June 09, 2022	Edit Delete
□	Kerero - lalesa - shall wel	Negelle Arsi - Kerero	Warui Francis	June 09, 2022	Edit Delete
□	Gude Dura - sotira - handpump	Negelle Arsi - Gude Dura	Warui Francis	June 09, 2022	Edit Delete
□	Kerero - lalesa - shall weell	Negelle Arsi - Kerero	Warui Francis	June 09, 2022	Edit Delete
□	Ali Wayo - Ali - deep wall	Negelle Arsi - Ali Wayo	Warui Francis	June 09, 2022	Edit Delete
□	Gude Dura - awulo mata arba - hand pump	Negelle Arsi - Gude Dura	Warui Francis	June 09, 2022	Edit Delete
□	Gebta Arjo - Gubeta Arjo - Water Kiosk-2	Negelle Arsi - Gebta Arjo	Warui Francis	May 25, 2022	Edit Delete
□	Gebta Arjo - Gubeta Arjo - Water Kiosk-1	Negelle Arsi - Gebta Arjo	Kabeto Abdo	May 25, 2022	Edit Delete
□	Gude Dura - Gode Duro - WP	Negelle Arsi - Gude Dura	Kabeto Abdo	May 25, 2022	Edit Delete
□	Kelo Tulu - Kelo Tulu - WP-3	Negelle Arsi - Kelo Tulu	Kabeto Abdo	May 25, 2022	Edit Delete

5.1.5 Editing data

A data entry can be edited.



Kerero - Adulala - shall weel

Location Demographics

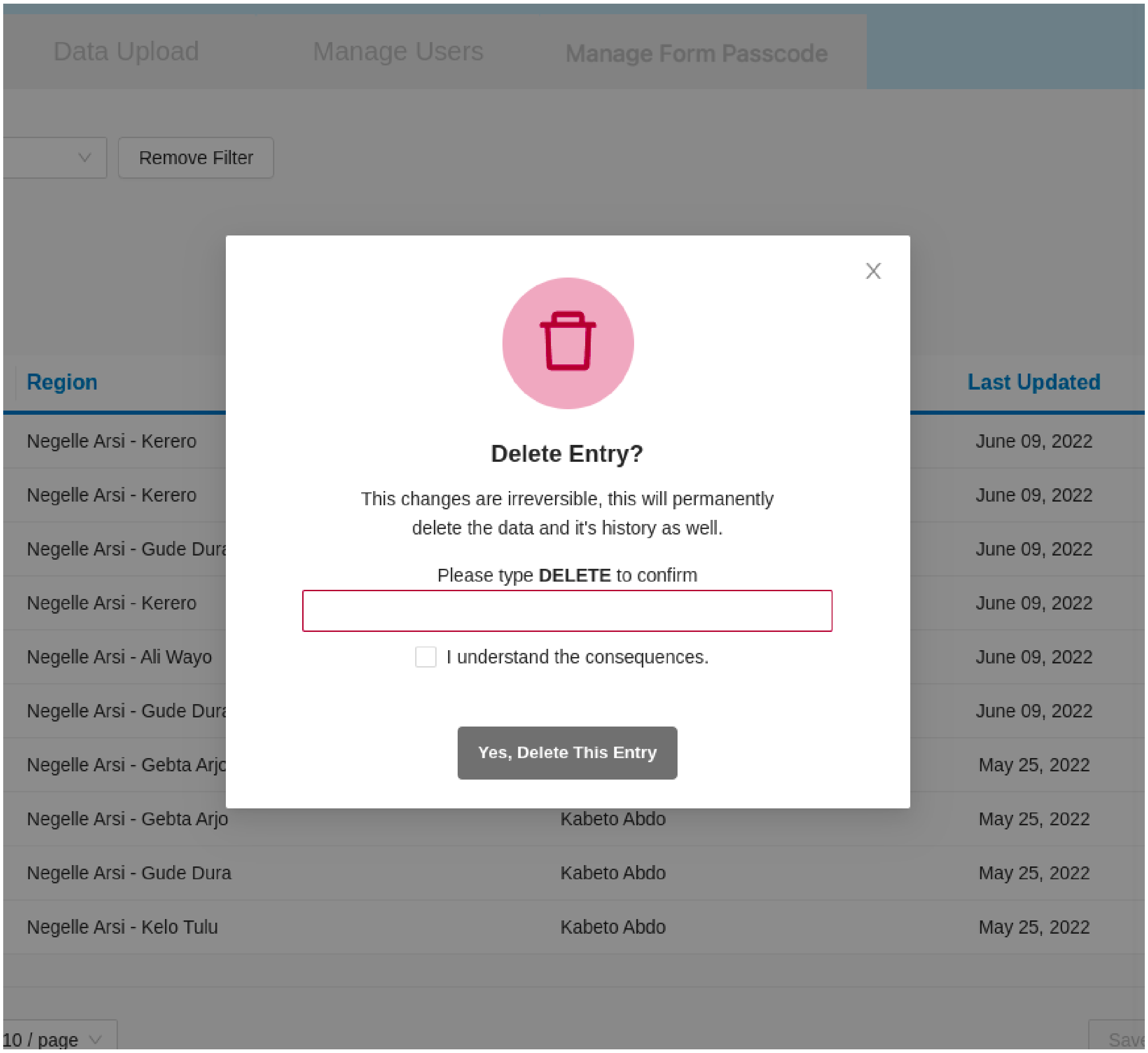
Location	Negelle Arsi - Kerero15
Name Of Village	Adulala
Gps Coordinates Of Village	**Lat: 7.48956939** 7.48956939|38.73544765 **Long: 38.73544765**

Water System Information

Water System Name	shall weel
Organization That Installed Water System	Catholic
Water Source Type	Shallow well
Depth Of Well M	60
Maximum Yield Lpd	3
Abstraction Pump Equipment	Solar system
Source Of Energy	Solar system
Number Of Functional Taps	1
Number Of Non	

5.1.6 Delete Data

A data entry can be deleted.



Region		Last Updated
Negelle Arsi - Kerero		June 09, 2022
Negelle Arsi - Kerero		June 09, 2022
Negelle Arsi - Gude Dur		June 09, 2022
Negelle Arsi - Kerero		June 09, 2022
Negelle Arsi - Ali Wayo		June 09, 2022
Negelle Arsi - Gude Dur		June 09, 2022
Negelle Arsi - Gebta Arj		May 25, 2022
Negelle Arsi - Gebta Arjo	Kabeto Abdo	May 25, 2022
Negelle Arsi - Gude Dura	Kabeto Abdo	May 25, 2022
Negelle Arsi - Kelo Tulu	Kabeto Abdo	May 25, 2022

Delete Entry?

This changes are irreversible, this will permanently delete the data and it's history as well.

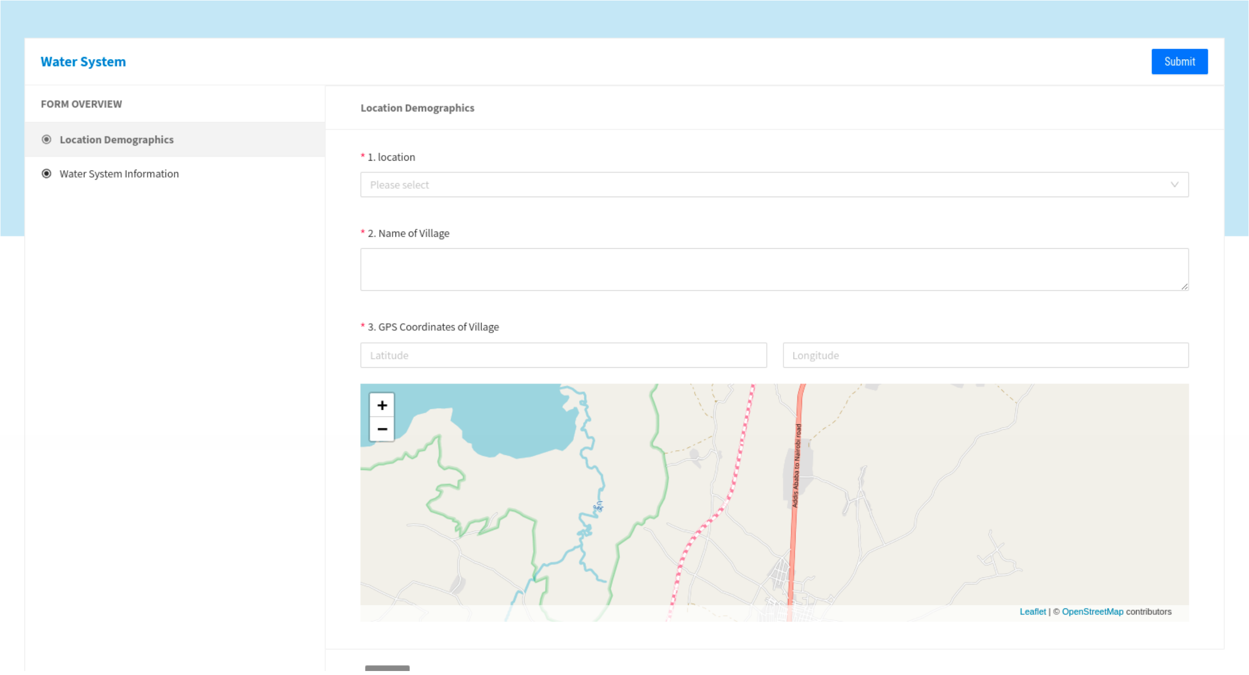
Please type **DELETE** to confirm

□ I understand the consequences.

Yes, Delete This Entry

5.1.7 Add New Data

You also can add a new data entry.



Location Demographics

* 1. location

Please select

* 2. Name of Village

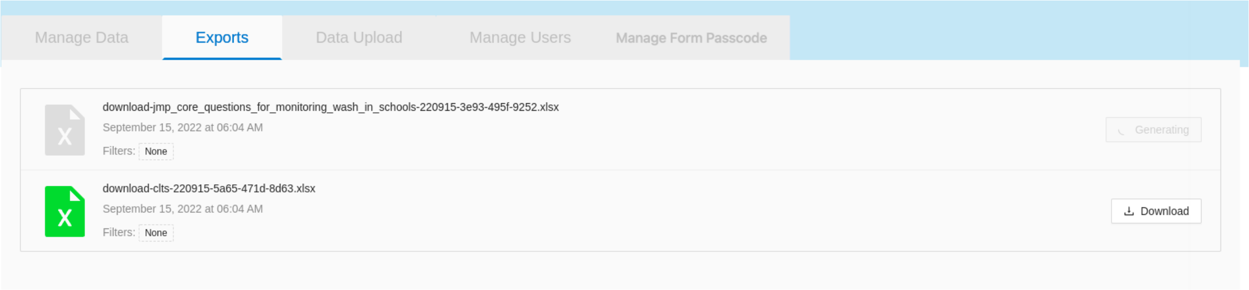
* 3. GPS Coordinates of Village

Latitude | Longitude

+
−
Addis Ababa to Nairobi road
Leaflet | © OpenStreetMap contributors

5.2 Export Data

You can see a list of exported data entries that you can download.

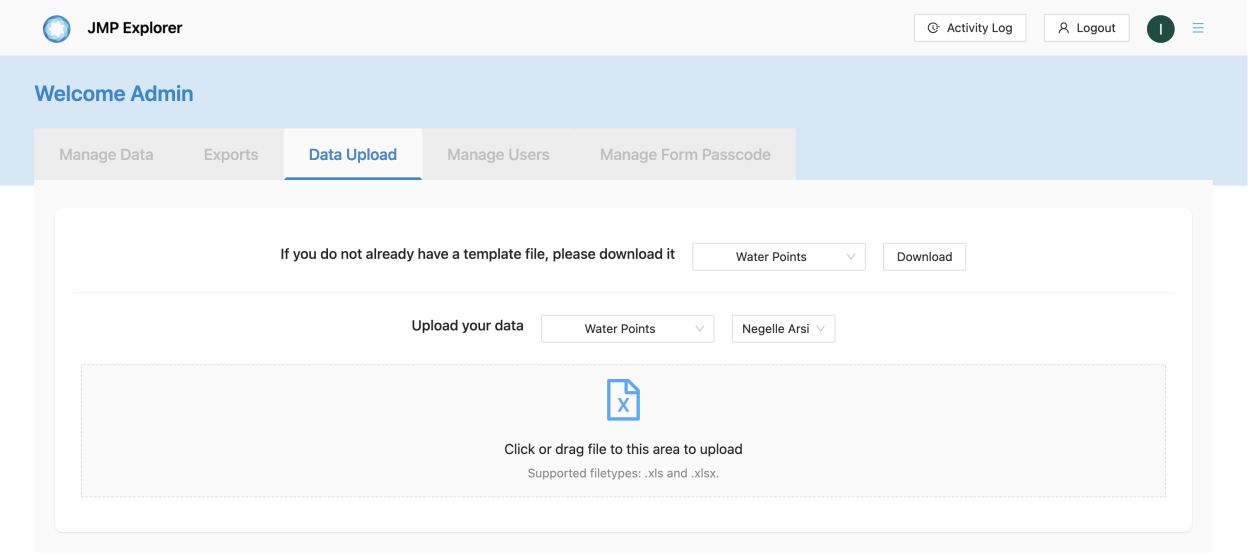


download-jmp_core_questions_for_monitoring_wash_in_schools-220915-3e93-495f-9252.xlsx
September 15, 2022 at 06:04 AM
Filters: None
Generating

download-clts-220915-5a65-471d-8d63.xlsx
September 15, 2022 at 06:04 AM
Filters: None
Download

5.3 Upload Data

You can also upload a new data entry from your **file manager** or download existing data by clicking on the **Download** button.



If you do not already have a template file, please download it

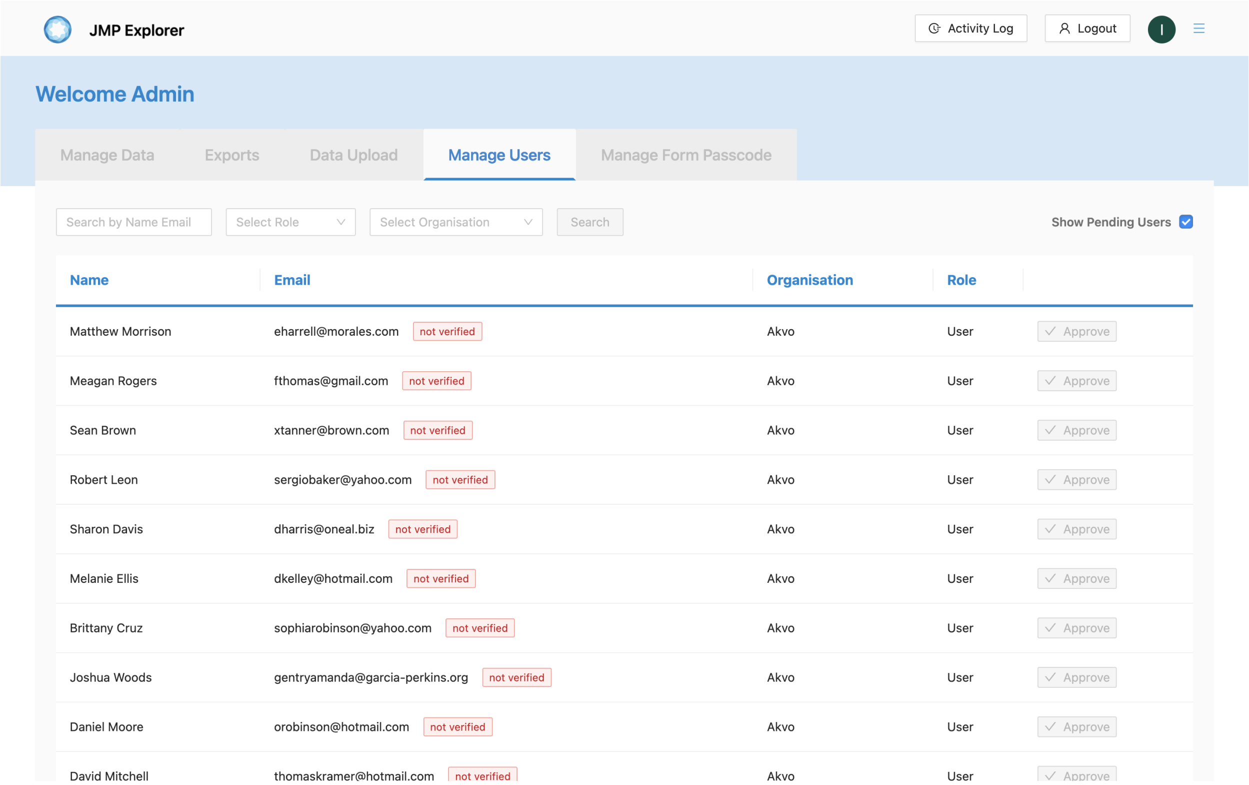
Upload your data

Click or drag file to this area to upload

Supported filetypes: .xls and .xlsx.

5.4 Manage users

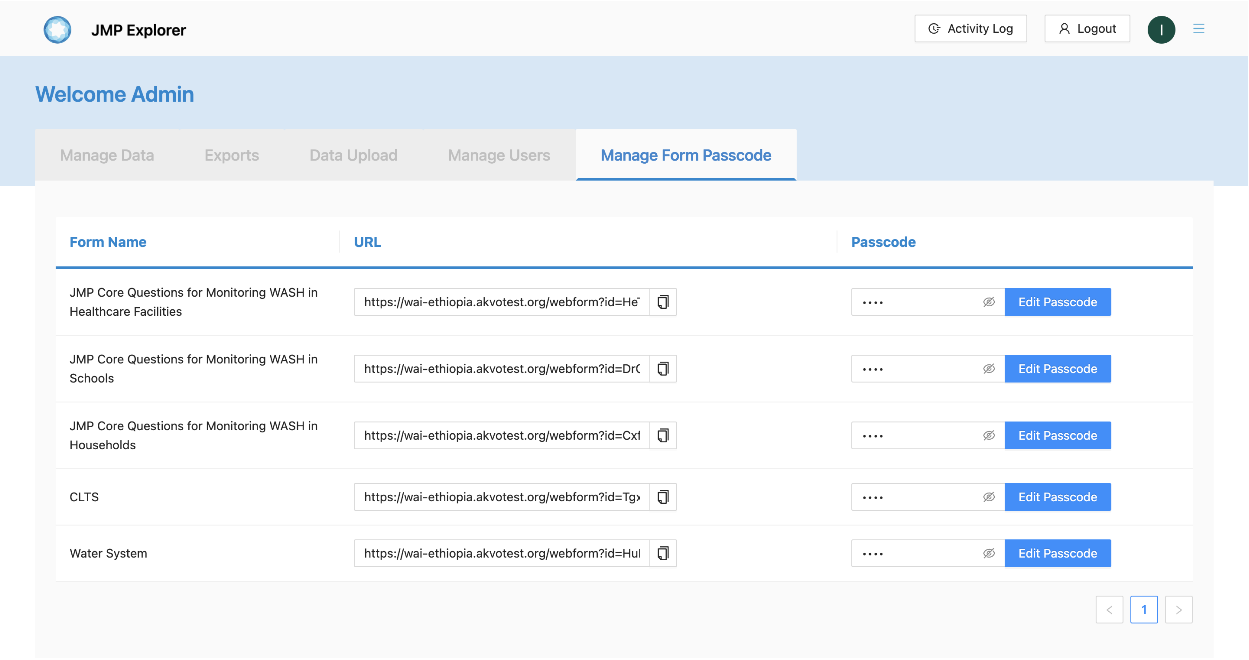
A list of users is displayed on the page with their: - Name - Email - Organisation - Role



Name	Email	Organisation	Role	
Matthew Morrison	eharrell@morales.com not verified	Akvo	User	✓ Approve
Meagan Rogers	fthomas@gmail.com not verified	Akvo	User	✓ Approve
Sean Brown	xtanner@brown.com not verified	Akvo	User	✓ Approve
Robert Leon	sergiobaker@yahoo.com not verified	Akvo	User	✓ Approve
Sharon Davis	dharris@oneal.biz not verified	Akvo	User	✓ Approve
Melanie Ellis	dkelley@hotmail.com not verified	Akvo	User	✓ Approve
Brittany Cruz	sophiarobinson@yahoo.com not verified	Akvo	User	✓ Approve
Joshua Woods	gentryamanda@garcia-perkins.org not verified	Akvo	User	✓ Approve
Daniel Moore	orobinson@hotmail.com not verified	Akvo	User	✓ Approve
David Mitchell	thomaskramer@hotmail.com not verified	Akvo	User	✓ Approve

5.5 Manage form passcode

A list of form that accessable via URL and is displayed on the page with the following columns: - Form name - URL - Passcode (The text field is encrypted by default)

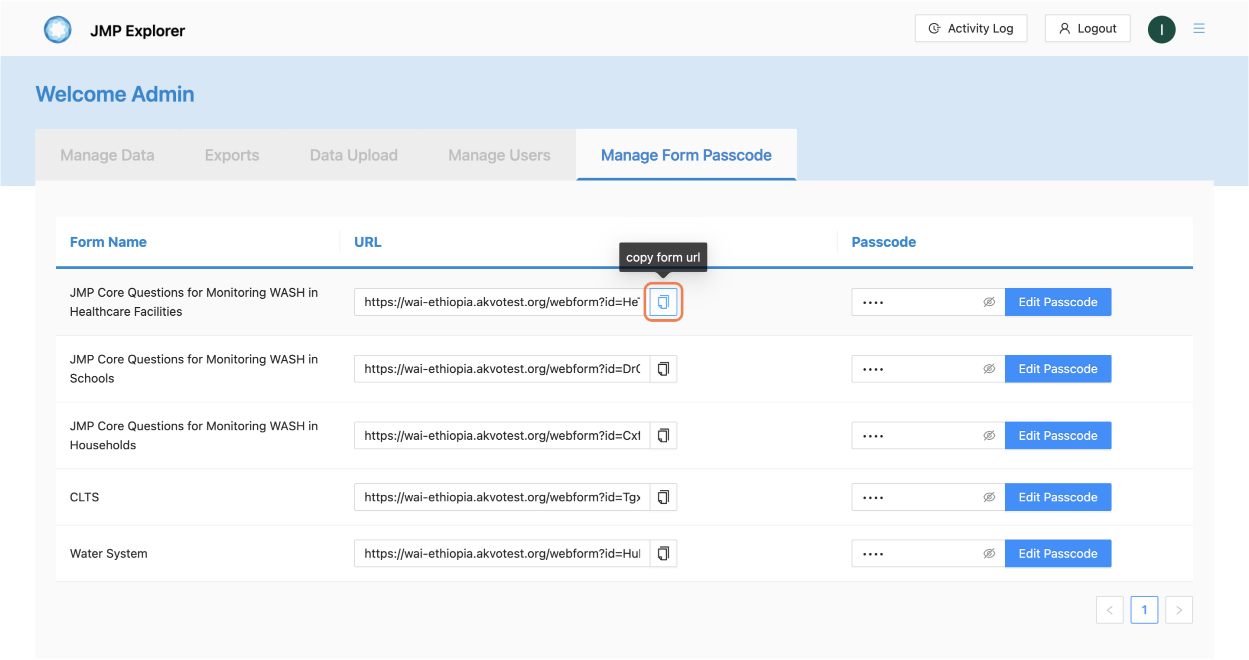


Welcome Admin

Form Name	URL	Passcode	
JMP Core Questions for Monitoring WASH in Healthcare Facilities	https://wai-ethiopia.akvotest.org/webform?id=He	••••	Edit Passcode
JMP Core Questions for Monitoring WASH in Schools	https://wai-ethiopia.akvotest.org/webform?id=Dr	••••	Edit Passcode
JMP Core Questions for Monitoring WASH in Households	https://wai-ethiopia.akvotest.org/webform?id=Cx	••••	Edit Passcode
CLTS	https://wai-ethiopia.akvotest.org/webform?id=Tg	••••	Edit Passcode
Water System	https://wai-ethiopia.akvotest.org/webform?id=Hu	••••	Edit Passcode

5.5.1 Copy URL

The URL field has a copy functionality to get URL easily



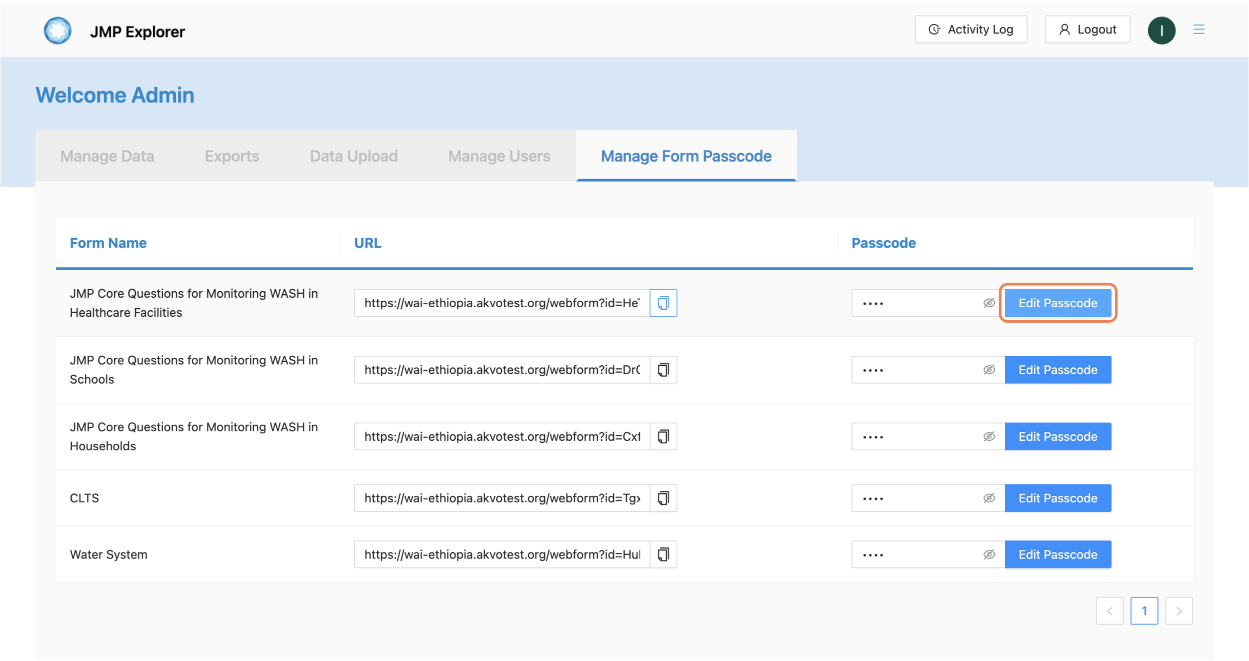
Welcome Admin

Manage Data | Exports | Data Upload | Manage Users | Manage Form Passcode

Form Name	URL	Passcode
JMP Core Questions for Monitoring WASH in Healthcare Facilities	https://wai-ethiopia.akvotest.org/webform?id=HeT	••••
JMP Core Questions for Monitoring WASH in Schools	https://wai-ethiopia.akvotest.org/webform?id=DrC	••••
JMP Core Questions for Monitoring WASH in Households	https://wai-ethiopia.akvotest.org/webform?id=Cxf	••••
CLTS	https://wai-ethiopia.akvotest.org/webform?id=Tgx	••••
Water System	https://wai-ethiopia.akvotest.org/webform?id=Hul	••••

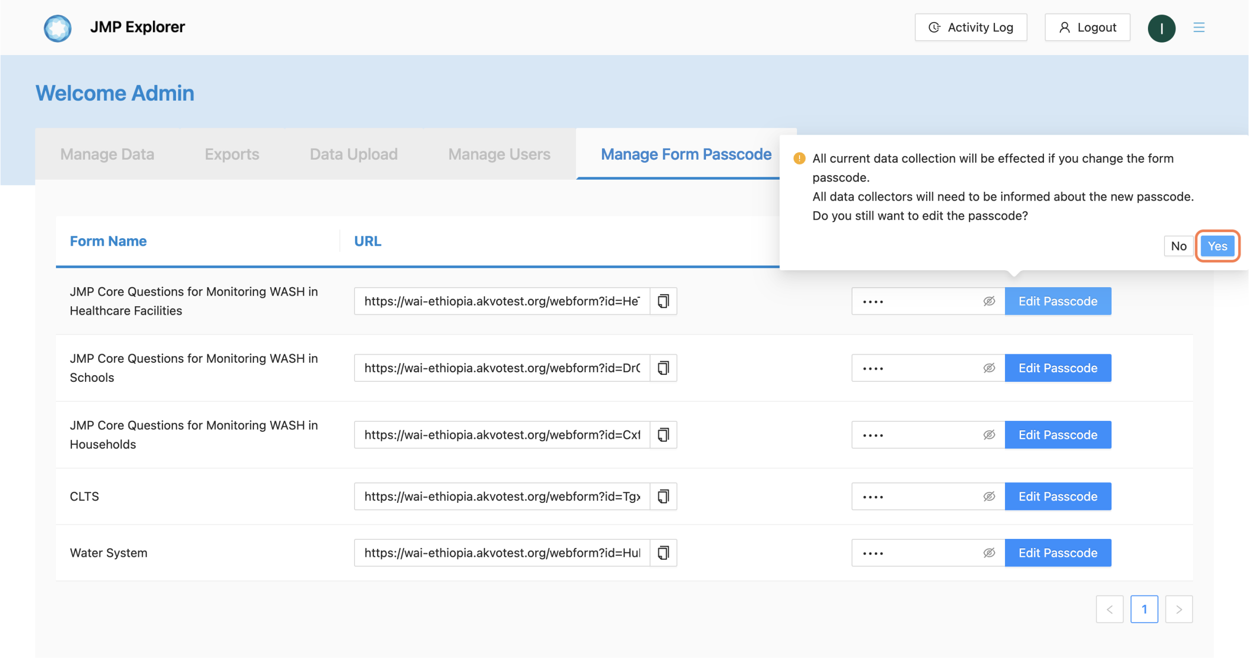
5.5.2 Edit Passcode

You can edit text field by clicking the **Edit passcode** button



Form Name	URL	Passcode
JMP Core Questions for Monitoring WASH in Healthcare Facilities	https://wai-ethiopia.akvotest.org/webform?id=He	•••• Edit Passcode
JMP Core Questions for Monitoring WASH in Schools	https://wai-ethiopia.akvotest.org/webform?id=Dr(	•••• Edit Passcode
JMP Core Questions for Monitoring WASH in Households	https://wai-ethiopia.akvotest.org/webform?id=Cxt	•••• Edit Passcode
CLTS	https://wai-ethiopia.akvotest.org/webform?id=Tg›	•••• Edit Passcode
Water System	https://wai-ethiopia.akvotest.org/webform?id=Hul	•••• Edit Passcode

Once edit is clicked, a warning pop-up will appear to confirm the user

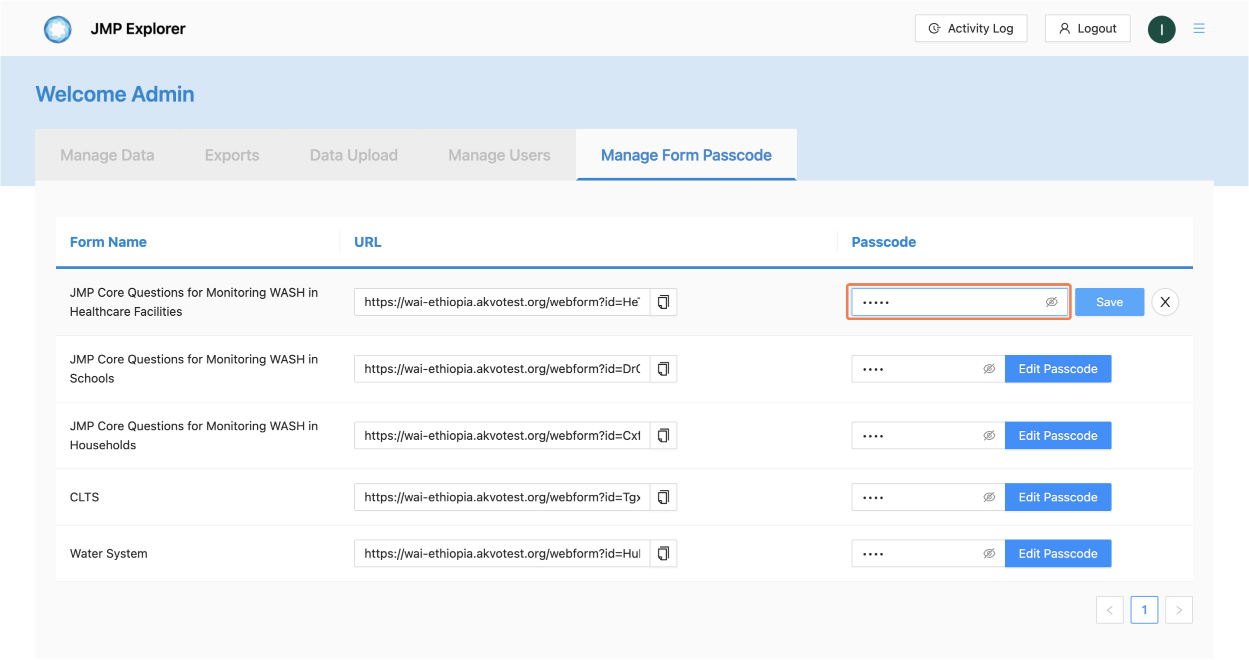


All current data collection will be effected if you change the form passcode.
All data collectors will need to be informed about the new passcode.
Do you still want to edit the passcode?

No Yes

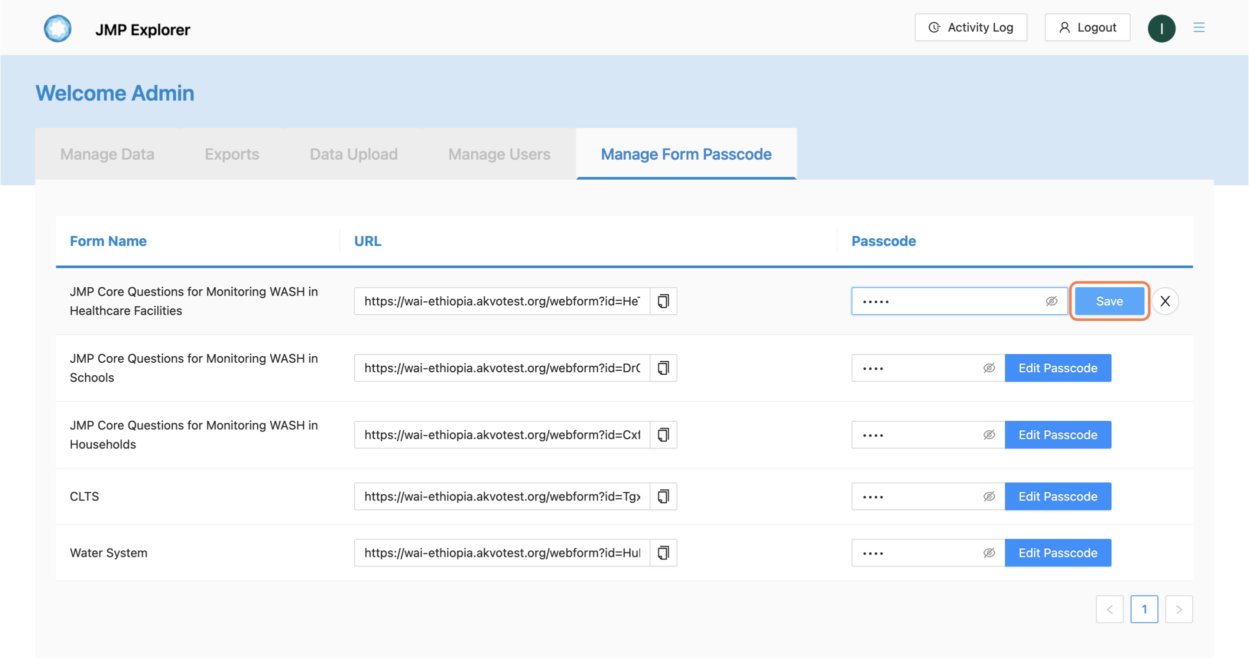
Form Name	URL	Passcode
JMP Core Questions for Monitoring WASH in Healthcare Facilities	https://wai-ethiopia.akvotest.org/webform?id=He	•••• Edit Passcode
JMP Core Questions for Monitoring WASH in Schools	https://wai-ethiopia.akvotest.org/webform?id=Dr(	•••• Edit Passcode
JMP Core Questions for Monitoring WASH in Households	https://wai-ethiopia.akvotest.org/webform?id=Cxt	•••• Edit Passcode
CLTS	https://wai-ethiopia.akvotest.org/webform?id=Tg›	•••• Edit Passcode
Water System	https://wai-ethiopia.akvotest.org/webform?id=Hul	•••• Edit Passcode

If you select yes, then the text field can be filled in with the new passcode. Otherwise, the text field will return readonly.



Form Name	URL	Passcode
JMP Core Questions for Monitoring WASH in Healthcare Facilities	https://wai-ethiopia.akvotest.org/webform?id=He	•••••
JMP Core Questions for Monitoring WASH in Schools	https://wai-ethiopia.akvotest.org/webform?id=Dr(	••••
JMP Core Questions for Monitoring WASH in Households	https://wai-ethiopia.akvotest.org/webform?id=Cxl	••••
CLTS	https://wai-ethiopia.akvotest.org/webform?id=Tg>	••••
Water System	https://wai-ethiopia.akvotest.org/webform?id=Hul	••••

Click the save button to save the passcode you just created



Form Name	URL	Passcode
JMP Core Questions for Monitoring WASH in Healthcare Facilities	https://wai-ethiopia.akvotest.org/webform?id=He	•••••
JMP Core Questions for Monitoring WASH in Schools	https://wai-ethiopia.akvotest.org/webform?id=Dr(	••••
JMP Core Questions for Monitoring WASH in Households	https://wai-ethiopia.akvotest.org/webform?id=Cxl	••••
CLTS	https://wai-ethiopia.akvotest.org/webform?id=Tg>	••••
Water System	https://wai-ethiopia.akvotest.org/webform?id=Hul	••••

CHAPTER

SIX

DEVELOPMENT INSTALLATION

6.1 Environment Setup

```
export WAI_AUTH0_DOMAIN="string_url"
export WAI_AUTH0_CLIENT_ID="string"
export WAI_AUTH0_SECRET="string"
export WAI_AUTH0_AUDIENCE="string"
export WAI_AUTH0_SECRET="string"
export WAI_AUTH0_SPA_DOMAIN="string_url"
export WAI_AUTH0_SPA_CLIENT_ID="string"
export INSTANCE_NAME="wai-demo"
```

6.2 Start the App

Once you have all the required environment ready, then run the App using:

- Run the application

  ```
  export INSTANCE_NAME=<project-name>
  docker compose up -d
  ```

- Stop

  ```
  docker compose down
  ```

- Reset the app

  ```
  docker compose down -v
  ```

CHAPTER

SEVEN

DATABASE SEEDER

7.1 Config Requirements

Before you seed the baseline data, please make sure that you have all the required file in the following structure:

Folder Path: */backend/source/*

```
/backend/source.
└── project-name
    ├── config.js
    ├── config.min.js
    ├── data
    │   └── organisation.csv
    ├── forms
    │   ├── 01-clts.json
    │   ├── 02-health.json
    │   ├── 03-hh.json
    │   ├── 04-school.json
    │   └── 05-wp.json
    └── topojson.js
```

7.1.1 config.json

config.min.js is pre-generated file to merge visualisation *config.js*, *topojson.js* and menu.

```
MINJS = jsmin("".join([
  "var levels=" + str([g["alias"] for g in GEO_CONFIG]) + ";"
  "var map_config={shapeLevels:" + str([g["name"]
                                      for g in GEO_CONFIG]) + "};",
  "var topojson=",
  open(f"{SOURCE_PATH}/topojson.json").read(),
  ";", JS_FILE, JS_i18n_FILE
]))
JS_FILE = f"{SOURCE_PATH}/config.min.js"
open(JS_FILE, 'w').write(MINJS)
```

7.1.2 forms.json

**.json* files inside forms folder is the form definition of a questionnaire which contains detail of forms including question group setting and question definition.

Example:

```
{
  "form": "JMP Core Questions for Monitoring WASH in Households",
  "id": 567420165,
  "question_groups": [
     {
       "question_group": "Location Demographics",
       "questions": [
         {
           "question": "location",
           "order": 1,
           "required": true,
           "type": "administration",
           "meta": true
         },
         {
           "id": 554110154,
           "question": "Village",
           "order": 3,
           "meta": true,
           "type": "text",
           "required": true,
           "options": null
         },
         {
           "id": 554110155,
           "question": "GPS Coordinates of Household",
           "order": 4,
           "meta": true,
           "type": "geo",
           "required": true,
           "options": null
         }
       ]
     }
   ]
 }
```

7.2 Seeder CLI

7.2.1 Administration Level Seeder

```
docker compose exec backend python -m seeder.administration
```

7.2.2 Organisation Seeder

```
docker compose exec backend python -m seeder.organisation
```

7.2.3 Super Admin

```
docker compose exec backend python -m seeder.admin youremail@akvo.org "Your Name" Akvo
```

7.2.4 Form Seeder

```
docker compose exec backend python -m seeder.form
```

7.2.5 Seed Fake User

```
docker compose exec backend python -m seeder.fake_user <number_of_user> Akvo
```

7.2.6 Datapoint Seeder

```
docker compose exec backend python -m seeder.fake_datapoint youremail@akvo.org <number_
↪of_datapoints>
```

CHAPTER

EIGHT

JMP LOGIC IMPLEMENTATION

JMP Logic implementation has been done by the AkvoResponseGrouper[1] library and we just need to create a category.json inside the source folder and place it in a specific instance.

For example, if we want to implement JMP logic on the Ethiopia instance, then category.json should be created on

```
backend
├── source
    ├── wai-ethiopia
        └── category.json
```

8.1 Properties

This section contains the properties that will be used in configuring the JMP logic in category.json.

8.1.1 Criteria's fields

Field	Type	Description
name	String	Criteria name
form	Integer	Existing form ID in database
categories	Array of category	List of categories

8.1.2 Category's fields

Field	Type	Description
name	String	Category name
questions	Array of question including the logic	List of existing questions and their logic

[1] AkvoResponseGrouper: https://pypi.org/project/AkvoResponseGrouper/

8.1.3 Question & logic's fields

Field	Type	Required	Description
id	Integer	Yes	Existing question ID in database.
text	String	No	Question description.
options	Array of string	Yes	Set list of options that will have intersections with the answer to the question.
other	Array of other	No	Another set of lists of options that don't have intersections in the *options*.
else	Object of *else*	No	Set category that has no intersections, either in *options* or *other*.

8.1.4 Other's fields

Field	Type	Required	Description
name	String	Yes	Category name
options	Array of string	Yes	Set list of options that will have intersections with the answer to the question
questions	Array of *question*	Yes	List of existing questions and their logic and can be set empty of Array *[]*

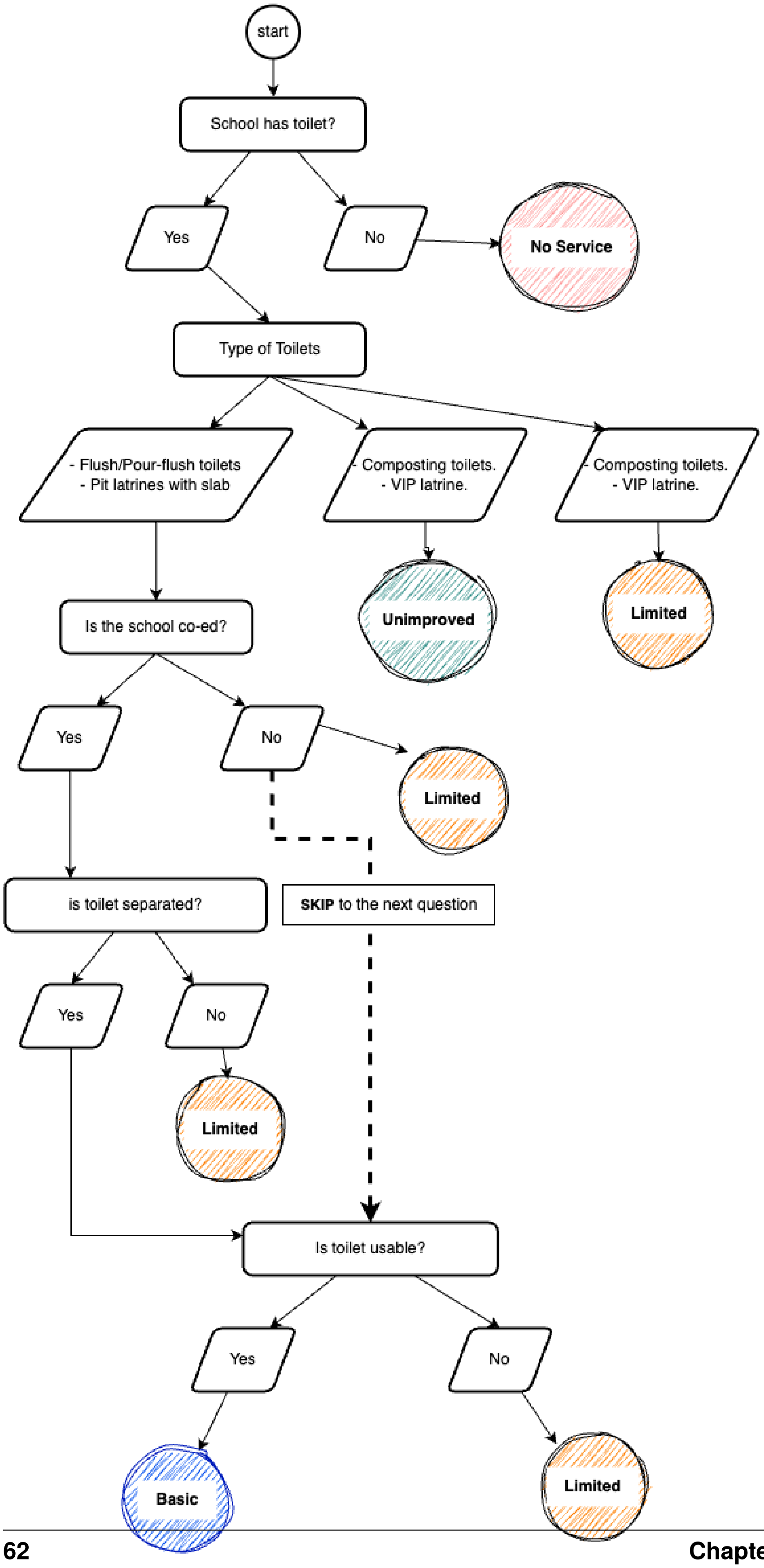
8.1.5 Else's fields

Field	Type	Required	Description
name	String	No	Category name
ignore	Array of existing question IDs	No	Set list of question IDs that can be skipped based on the intersections in the options

8.2 Example

In this section, we provide an example use case to demonstrate how to create a category.json file. Please note that the presented use case, "Sanitation," is intended for illustrative purposes only. While the example showcases the functionality and features of our library, it may not be an exact representation of real-world scenarios.

8.2.1 Logic visualisation



8.2.2 JSON File (category.json)

```
[
  {
    "name": "Sanitation Criteria",
    "form": 1,
    "categories": [
      {
        "name": "Basic",
        "questions": [
          {
              "id": 11,
              "text": "School has toilet?",
              "options": [
                  "Yes"
              ],
              "else": {
                  "name": "No service"
              }
          },
          {
              "id": 12,
              "text": "Type of toilets",
              "options": [
                  "Flush/Pour-flush toilets",
                  "Pit latrines with slab"
              ],
              "other": [
                  {
                      "name": "Unimproved",
                      "options": [
                          "Composting toilets",
                          "VIP latrine"
                      ],
                      "questions": []
                  }
              ],
              "else": {
                  "name": "Limited"
              }
          },
          {
              "id": 13,
              "text": "Is the school co-ed?",
              "options": [
                  "Yes"
              ],
              "else": {
                  "ignore": [
                      14
                  ]
              }
          },
```

(continues on next page)

(continued from previous page)

```
          {
              "id": 14,
              "text": "is toilet separated?",
              "options": [
                  "Yes"
              ],
              "else": {
                  "name": "Limited"
              }
          },
          {
              "id": 15,
              "text: ":"Is toilet usable?",
              "options": [
                  "Yes"
              ],
              "else": {
                  "name": "Limited"
              }
          }
        ]
      }
    ]
  }
]
```

CHAPTER

NINE

RUNNING TESTS

9.1 Backend Test

```
docker-compose exec backend ./test.sh
```

9.2 Frontend Test

```
docker-compose exec frontend yarn test
```

CHAPTER

TEN

DATABASE SCHEMA

10.1 List of Table

```
SELECT relname, relkind
FROM   pg_class
WHERE relreplident = 'd'
AND relhasindex = true;
```

Table 1: List of Table

relname	relkind
alembic_version	r
administration	r
access	r
question_group	r
form	r
question	r
data	r
answer	r
organisation	r
jobs	r
log	r
option	r
history	r
user	r

10.2 Table Relationship

Table 2: List of Relationship

Schema	Name	Type	Owner
public	access	table	wai
public	access_id_seq	sequence	wai
public	administration	table	wai
public	administration_id_seq	sequence	wai
public	alembic_version	table	wai
public	answer	table	wai
public	answer_id_seq	sequence	wai
public	answer_search	view	wai
public	data	table	wai
public	data_id_seq	sequence	wai
public	form	table	wai
public	form_id_seq	sequence	wai
public	history	table	wai
public	history_id_seq	sequence	wai
public	jobs	table	wai
public	jobs_id_seq	sequence	wai
public	log	table	wai
public	log_id_seq	sequence	wai
public	option	table	wai
public	option_id_seq	sequence	wai
public	organisation	table	wai
public	organisation_id_seq	sequence	wai
public	question	table	wai
public	question_group	table	wai
public	question_group_id_seq	sequence	wai
public	question_id_seq	sequence	wai
public	score_view	view	wai
public	user	table	wai
public	user_id_seq	sequence	wai

10.3 Table Details

10.3.1 Administration

```
SELECT ordinal_position as pos, column_name, data_type, column_default, is_
↪nullable
FROM   information_schema.columns
WHERE  table_name = 'administration'
ORDER  BY ordinal_position;
```

Table 3: List of Relationship

pos	column_name	data_type	udt_name	column_default	is_nullable
1	id	integer	int4	nextval(‘administration_id_seq’::regclass)	NO
2	parent	integer	int4		YES
3	name	character varying	varchar		YES

10.3.2 User

```
SELECT ordinal_position as pos, column_name, data_type, column_default, is_
↪nullable
FROM   information_schema.columns
WHERE  table_name = 'user'
ORDER  BY ordinal_position;
```

Table 4: User Table

pos	column_name	data_type	column_default	is_nullable
1	id	integer	nextval(‘user_id_seq’::regclass)	NO
2	email	character varying		YES
3	active	boolean		YES
4	role	USER-DEFINED		YES
5	created	timestamp without time zone		YES
6	organisation	integer		YES
7	name	character varying		YES
8	__ts_vector__	tsvector		YES
9	manage_form_passcode	boolean	false	NO

10.3.3 User Access

```
SELECT ordinal_position as pos, column_name, data_type, column_default, is_
↪nullable
FROM   information_schema.columns
WHERE  table_name = 'access'
ORDER  BY ordinal_position;
```

Table 5: Access Table

pos	column_name	data_type	column_default	is_nullable
1	id	integer	nextval(‘access_id_seq’::regclass)	NO
2	user	integer		YES
3	administration	integer		YES

10.3.4 Organisation

```
SELECT ordinal_position as pos, column_name, data_type, column_default, is_
↪nullable
FROM   information_schema.columns
WHERE  table_name = 'organisation'
ORDER  BY ordinal_position;
```

Table 6: Organisation Table

pos	column_name	data_type	column_default	is_nullable
1	id	integer	nextval(‘organisation_id_seq’::regclass)	NO
2	name	character varying		YES
3	type	USER-DEFINED		YES
4	created	timestamp without time zone		YES

10.3.5 Form

```
SELECT ordinal_position as pos, column_name, data_type, column_default, is_
↪nullable
FROM   information_schema.columns
WHERE  table_name = 'form'
ORDER  BY ordinal_position;
```

Table 7: Form Table

pos	column_name	data_type	column_default	is_nullable
1	id	integer	nextval(‘form_id_seq’::regclass)	NO
2	name	character varying		YES
3	description	text		YES
4	default_language	character varying		YES
5	languages	ARRAY		YES
6	translations	ARRAY		YES
7	version	double precision		YES

10.3.6 Question Group

```
SELECT ordinal_position as pos, column_name, data_type, column_default, is_
↪nullable
FROM   information_schema.columns
WHERE  table_name = 'question_group'
ORDER  BY ordinal_position;
```

Table 8: Question Group Table

pos	column_name	data_type	column_default	is_nullable
1	id	integer	nextval('question_group_id_seq'::regclass)	NO
2	order	integer		YES
3	name	character varying		YES
4	form	integer		YES
5	description	text		YES
6	repeatable	boolean	false	YES
7	repeat_text	character varying		YES
8	translations	ARRAY		YES

10.3.7 Question

```
SELECT ordinal_position as pos, column_name, data_type, column_default, is_
↪nullable
FROM   information_schema.columns
WHERE  table_name = 'question'
ORDER  BY ordinal_position;
```

Table 9: Question Table

pos	column_name	data_type	column_default	is_nullable
1	id	integer	nextval('question_id_seq'::regclass)	NO
2	order	integer		YES
3	name	character varying		YES
4	form	integer		YES
5	meta	boolean		NO
6	type	USER-DEFINED		YES
7	question_group	integer		YES
8	required	boolean	true	NO
9	rule	jsonb		YES
10	dependency	ARRAY		YES
11	tooltip	jsonb		YES
12	translations	ARRAY		YES
13	api	jsonb		YES
14	addons	jsonb		YES

10.3.8 Question Option

```
SELECT ordinal_position as pos, column_name, data_type, column_default, is_
↪nullable
FROM   information_schema.columns
WHERE  table_name = 'option'
ORDER  BY ordinal_position;
```

Table 10: Question Option Table

pos	column_name	data_type	column_default	is_nullable
1	id	integer	nextval('option_id_seq'::regclass)	NO
2	order	integer		YES
3	name	character varying		YES
4	question	integer		YES
5	color	character varying		YES
6	score	integer		YES
7	code	character varying		YES
8	translations	ARRAY		YES

10.3.9 Data

```
SELECT ordinal_position as pos, column_name, data_type, column_default, is_
↪nullable
FROM   information_schema.columns
WHERE  table_name = 'data'
ORDER  BY ordinal_position;
```

Table 11: Data Table

pos	column_name	data_type	column_default	is_nullable
1	id	integer	nextval('data_id_seq'::regclass)	NO
2	name	character varying		YES
3	form	integer		YES
4	administration	integer		YES
5	geo	ARRAY		YES
6	created_by	integer		YES
7	updated_by	integer		YES
8	created	timestamp without time zone	CURRENT_TIMESTAMP	YES
9	updated	timestamp without time zone		YES

10.3.10 Data Answer

```
SELECT ordinal_position as pos, column_name, data_type, column_default, is_
↪nullable
FROM   information_schema.columns
WHERE  table_name = 'answer'
ORDER  BY ordinal_position;
```

Table 12: Data Answer Table

pos	column_name	data_type	column_default	is_nullable
1	id	integer	nextval('answer_id_seq'::regclass)	NO
2	question	integer		YES
3	data	integer		YES
4	value	double precision		YES
5	text	text		YES
6	options	ARRAY		YES
7	created_by	integer		YES
8	updated_by	integer		YES
9	created	timestamp without time zone	CURRENT_TIMESTAMP	YES
10	updated	timestamp without time zone		YES

10.3.11 Data History

```
SELECT ordinal_position as pos, column_name, data_type, column_default, is_
↪nullable
FROM   information_schema.columns
WHERE  table_name = 'history'
ORDER  BY ordinal_position;
```

Table 13: Data Answer Table

pos	column_name	data_type	column_default	is_nullable
1	id	integer	nextval('history_id_seq'::regclass)	NO
2	question	integer		YES
3	data	integer		YES
4	value	double precision		YES
5	text	text		YES
6	options	ARRAY		YES
7	created_by	integer		YES
8	updated_by	integer		YES
9	created	timestamp without time zone	CURRENT_TIMESTAMP	YES
10	updated	timestamp without time zone		YES

10.3.12 Jobs

```
SELECT ordinal_position as pos, column_name, data_type, column_default, is_
↪nullable
FROM   information_schema.columns
WHERE  table_name = 'jobs'
ORDER  BY ordinal_position;
```

Table 14: Jobs Table

pos	column_name	data_type	column_default	is_nullable
1	id	integer	nextval('jobs_id_seq'::regclass)	NO
2	type	USER-DEFINED		YES
3	status	USER-DEFINED	'pending'::jobstatus	YES
4	payload	text		NO
5	info	jsonb		YES
6	attempt	integer	0	YES
7	created_by	integer		NO
8	created	timestamp without time zone	CURRENT_TIMESTAMP	YES
9	available	timestamp without time zone		YES

10.3.13 Logs

```
SELECT ordinal_position as pos, column_name, data_type, column_default, is_
↪nullable
FROM   information_schema.columns
WHERE  table_name = 'log'
ORDER  BY ordinal_position;
```

Table 15: Log Table

pos	column_name	data_type	column_default	is_nullable
1	id	integer	nextval('log_id_seq'::regclass)	NO
2	user	integer		YES
3	message	text		YES
4	at	timestamp without time zone	CURRENT_TIMESTAMP	YES
5	jobs	integer		YES

CHAPTER ELEVEN

LIST OF SERVICES

```
** Services

#+NAME: Services
#+begin_src sh :results verbatim output :exports both
docker compose ps
#+end_src

#+RESULTS: Services
: NAME                             IMAGE                                              ␣
↪COMMAND                   SERVICE             CREATED             STATUS              ␣
↪PORTS
: wai-sdg-portal-backend-1       python:3.8.5                                        "./
↪dev.sh"                backend             4 hours ago         Up 4 hours
: wai-sdg-portal-db-1            postgres:12-alpine
↪"docker-entrypoint.s..."   db                  4 hours ago         Up 4 hours          ␣
↪0.0.0.0:5432->5432/tcp, :::5432->5432/tcp
: wai-sdg-portal-frontend-1      akvo/akvo-node-18-alpine:20220923.084347.0558ee6   "run-
↪as-user.sh ./st..."    frontend            4 hours ago         Up 4 hours
: wai-sdg-portal-mainnetwork-1   alpine:3.14.0
↪"tail -f /dev/null"      mainnetwork         4 hours ago         Up 4 hours          0.
↪0.0.0:3000->3000/tcp, :::3000->3000/tcp, 0.0.0.0:5050->5050/tcp, :::5050->5050/tcp, 0.
↪0.0.0:8000->8000/tcp, :::8000->8000/tcp
: wai-sdg-portal-pgadmin-1       dpage/pgadmin4:5.7                                  "/
↪entrypoint.sh"          pgadmin             4 hours ago         Up 4 hours
: wai-sdg-portal-worker-1        python:3.8.5                                        "./
↪worker.sh"             worker              4 hours ago         Up 4 hours

** Container Detail

*** Backend

#+NAME: Backend Processes
#+begin_src sh :exports both
docker compose top backend | tail -n +2 | head -n -1
#+end_src

#+RESULTS: Backend Processes
| UID  |   PID |  PPID | C | STIME | TTY |     TIME | CMD                   |           ␣
↪             |          |                                   |        |              |   ␣
↪                       |                |                         |
```

(continues on next page)

(continued from previous page)

```
| root | 94628 | 94577 | 0 | 12:12 | ?   | 00:00:00 | bash                  | ./dev.sh  ␣
↪             |          |                                  |        |              |   ␣
↪                  |                |                        |
| root | 95594 | 94628 | 3 | 12:12 | ?   | 00:08:17 | /usr/local/bin/python | /usr/local/
↪bin/uvicorn | main:app | --reload                         | --port | 5000          |   ␣
↪                  |                |                        |
| root | 95595 | 95594 | 0 | 12:12 | ?   | 00:00:00 | /usr/local/bin/python | -c        ␣
↪             | from      | multiprocessing.resource_tracker | import | main;main(4) |   ␣
↪                  |                |                        |
| root | 95596 | 95594 | 0 | 12:12 | ?   | 00:00:49 | /usr/local/bin/python | -c        ␣
↪             | from      | multiprocessing.spawn            | import | spawn_main;  |␣
↪spawn_main(tracker_fd=5, | pipe_handle=7) | --multiprocessing-fork |

**** Commands

#+NAME: Backend Commands
#+begin_src sh :results verbatim :exports both
docker compose exec backend ./seed.sh
#+end_src

#+RESULTS: Backend Commands
: This script require more than 0 argument/s
: Example: ./test.sh dev@akvo.org "My Name" "My Organisation"

#+NAME: Seeder
#+begin_src sh :results verbatim :exports both
docker compose exec backend cat ./seed.sh | grep seeder. | sed 's/#\ //g'
#+end_src

#+RESULTS: Seeder
:     python -m seeder.administration
:     python -m seeder.admin "$@"
:     python -m seeder.fake_user 30 "$3"
:     python -m seeder.form
:     python -m seeder.datapoint "$1"

*** Frontend

**** Processes

#+NAME: Frontend Processes
#+begin_src sh :exports both
docker compose -f docker-compose.yml top frontend | tail -n +2 | head -n -1
#+end_src

#+RESULTS: Frontend Processes
| UID  |   PID |  PPID | C | STIME | TTY |     TIME | CMD                     |           ␣
↪                                      |       |
| root | 94620 | 94559 | 0 | 12:12 | ?   | 00:00:00 | bash                    | ./start.sh  ␣
↪                                      |       |
| root | 94820 | 94620 | 0 | 12:12 | ?   | 00:00:03 | node                    | /opt/yarn-v1.
↪22.19/bin/yarn.js                     | start |
```

(continues on next page)

(continued from previous page)

```
| root | 94841 | 94820 | 0 | 12:12 | ?   | 00:00:00 | /usr/local/bin/node | /app/node_
↪modules/.bin/react-scripts            | start |
| root | 94849 | 94841 | 0 | 12:12 | ?   | 00:00:21 | /usr/local/bin/node | /app/node_
↪modules/react-scripts/scripts/start.js |       |

**** Commands

#+NAME: Frontend Commands
#+begin_src sh :results verbatim :exports both
docker compose exec frontend yarn run
#+end_src

#+RESULTS: Frontend Commands
#+begin_example
yarn run v1.22.19
info Commands available from binary scripts: acorn, ansi-html, autoprefixer,␣
↪browserslist, browserslist-lint, cross-env, cross-env-shell, css-blank-pseudo, css-has-
↪pseudo, css-prefers-color-scheme, cssesc, detect, detect-port, detective, ejs,␣
↪escodegen, esgenerate, eslint, eslint-config-prettier, esparse, esvalidate, he, html-
↪minifier-terser, import-local-fixture, is-docker, jake, jest, js-yaml, jsesc, json5,␣
↪loose-envify, lz-string, mime, mkdirp, multicast-dns, nanoid, nmtree, node-which,␣
↪parser, prettier, react-scripts, regjsparser, remarkable, resolve, rimraf, rollup,␣
↪sass, semver, sha.js, svgo, synp, tailwind, tailwindcss, terser, topo2geo, topomerge,␣
↪topoquantize, uuid, webpack, webpack-dev-server, yarn-audit-fix
info Project commands
   - build
      react-scripts build
   - eject
      react-scripts eject
   - lint
      eslint --config .eslintrc.json src --ext .js,.jsx
   - prettier
      prettier --check src
   - start
      react-scripts start
   - test
      react-scripts test --updateSnapshot --transformIgnorePatterns "node_modules/(?!
↪d3|d3-geo|d3-array|internmap|delaunator|robust-predicates|react-leaflet)/"
   - test:ci
      CI=true react-scripts test --watchAll=false --coverage --transformIgnorePatterns
↪"node_modules/(?!d3|d3-geo|d3-array|internmap|delaunator|robust-predicates)/"
Done in 0.02s.
#+end_example

*** Worker

#+NAME: Worker Processes
#+begin_src sh :exports both
docker compose top worker | tail -n +2 | head -n -1
#+end_src

#+RESULTS: Worker Processes
```

(continues on next page)

(continued from previous page)

```
| UID  |   PID |  PPID | C | STIME | TTY |     TIME | CMD                    |         ␣
↪            |                |                                  |        |         ␣
↪|                          |                |                        |
| root | 94704 | 94679 | 0 | 12:12 | ?   | 00:00:00 | bash                   | ./worker.
↪sh            |                |                                  |        |         ␣
↪  |                          |                |                        |
| root | 95588 | 94704 | 3 | 12:12 | ?   | 00:08:19 | /usr/local/bin/python | /usr/local/
↪bin/uvicorn | worker:worker | --reload                           | --port | 5001      ␣
↪|                          |                |                        |
| root | 95590 | 95588 | 0 | 12:12 | ?   | 00:00:00 | /usr/local/bin/python | -c        ␣
↪            | from           | multiprocessing.resource_tracker | import | main;main(4)␣
↪|                          |                |                        |
| root | 95591 | 95588 | 0 | 12:12 | ?   | 00:00:55 | /usr/local/bin/python | -c        ␣
↪            | from           | multiprocessing.spawn            | import | spawn_main; ␣
↪| spawn_main(tracker_fd=5, | pipe_handle=7) | --multiprocessing-fork |

*** Database

#+NAME: Database Processes
#+begin_src sh :exports both
docker compose top db | tail -n +2 | head -n -1
#+end_src

#+RESULTS: Database Processes
| UID |    PID |  PPID | C | STIME | TTY |     TIME | CMD       |               |         ␣
↪    |                   |       |
|  70 |  94342 | 94316 | 0 | 12:12 | ?   | 00:00:00 | postgres  |               |         ␣
↪    |                   |       |
|  70 |  94635 | 94342 | 0 | 12:12 | ?   | 00:00:00 | postgres: | checkpointer |         ␣
↪    |                   |       |
|  70 |  94636 | 94342 | 0 | 12:12 | ?   | 00:00:00 | postgres: | background    | writer ␣
↪    |                   |       |
|  70 |  94637 | 94342 | 0 | 12:12 | ?   | 00:00:00 | postgres: | walwriter     |         ␣
↪    |                   |       |
|  70 |  94638 | 94342 | 0 | 12:12 | ?   | 00:00:00 | postgres: | autovacuum    |␣
↪launcher    |                   |       |
|  70 |  94639 | 94342 | 0 | 12:12 | ?   | 00:00:00 | postgres: | stats         |␣
↪collector   |                   |       |
|  70 |  94640 | 94342 | 0 | 12:12 | ?   | 00:00:00 | postgres: | logical       |␣
↪replication |          launcher |       |
|  70 |  95628 | 94342 | 0 | 12:12 | ?   | 00:00:00 | postgres: | wai           | wai_
↪demo     | 172.20.0.3(38228) | idle |
|  70 |  95629 | 94342 | 0 | 12:12 | ?   | 00:00:02 | postgres: | wai           | wai_
↪demo     | 172.20.0.3(38234) | idle |
|  70 |  95642 | 94342 | 0 | 12:12 | ?   | 00:00:00 | postgres: | wai           | wai_
↪demo     | 172.20.0.3(38242) | idle |
|  70 | 103242 | 94342 | 0 | 12:17 | ?   | 00:00:00 | postgres: | wai           | wai_
↪demo     | 172.20.0.3(36900) | idle |
|  70 | 103243 | 94342 | 0 | 12:17 | ?   | 00:00:00 | postgres: | wai           | wai_
↪demo     | 172.20.0.3(36916) | idle |
|  70 | 103244 | 94342 | 0 | 12:17 | ?   | 00:00:00 | postgres: | wai           | wai_
↪demo     | 172.20.0.3(36926) | idle |
```

(continues on next page)

(continued from previous page)

```
*** Dev

**** PG Admin

#+NAME: PG Admin Processes
#+begin_src sh :exports both
docker compose top pgadmin | tail -n +2 | head -n -1
#+end_src

#+RESULTS: PG Admin Processes
| UID      |    PID |  PPID | C | STIME | TTY |     TIME | CMD                         |␣
↪                   |           |       |        |           |    |   |           |   ␣
↪|                  |   |    |                    |                 |
| 5050     |  94608 | 94537 | 0 | 12:12 | ?   | 00:00:03 | /venv/bin/python3           |␣
↪/venv/bin/gunicorn | --timeout | 86400 | --bind | [::]:5050 | -w | 1 | --threads | 25␣
↪| --access-logfile | - | -c | gunicorn_config.py | run_pgadmin:app |
| root     |  94812 | 94608 | 0 | 12:12 | ?   | 00:00:00 | /usr/libexec/postfix/master |␣
↪-w                 |           |       |        |           |    |   |           |   ␣
↪|                  |   |    |                    |                 |
| systemd+ |  94814 | 94812 | 0 | 12:12 | ?   | 00:00:00 | qmgr                        |␣
↪-l                 | -t        |  unix | -u     |           |    |   |           |   ␣
↪|                  |   |    |                    |                 |
| 5050     |  94905 | 94608 | 0 | 12:12 | ?   | 00:00:09 | /venv/bin/python3           |␣
↪/venv/bin/gunicorn | --timeout | 86400 | --bind | [::]:5050 | -w | 1 | --threads | 25␣
↪| --access-logfile | - | -c | gunicorn_config.py | run_pgadmin:app |
| systemd+ | 425845 | 94812 | 0 | 15:32 | ?   | 00:00:00 | pickup                      |␣
↪-l                 | -t        |  unix | -u     |           |    |   |           |   ␣
↪|                  |   |    |                    |                 |

**** Main Network

#+NAME: Main Network Processes
#+begin_src sh :exports both
docker compose top mainnetwork | tail -n +2 | head -n -1
#+end_src

#+RESULTS: Main Network Processes
| UID  |   PID |  PPID | C | STIME | TTY |     TIME | CMD  |    |           |
| root | 94393 | 94363 | 0 | 12:12 | ?   | 00:00:00 | tail | -f | /dev/null |
```