# User Guide

## Introduction

This guide intends to show users how to how to install, create an incident through the Dossa mobile application, and other functions of the Dossier Management System web application. The following system design in Figure 1, show the Dossier Management System as collection of microservices working in addendum with each other’s.

Graphical user interface

Description automatically generated

Figure 1. System design of communication flow between users and backend services

## Installation and Using the Application

The Dossier Management System consist of five subsystems: VPN Server using Wireguard, the database system, the backend, web application, and a mobile application.

## Download and install requisite tools:

Download and Install Docker. Installation guide for Docker is available here: <https://docs.docker.com/desktop/>

In the Command Terminal, verify that Docker is installed with the following command:

docker --version

A properly installed docker would display the docker version and build number.

docker version 20.10.10, build b485636

Download and Install Git. Installation guide for GIT is available here: <https://git-scm.com/downloads>

In the Command Terminal, verify that GIT is installed with the following command:

git --version

A properly installed git would display the git version and build number.

git version 2.19.1.windows.1

### Install and run the Wireguard VPN Server (Recommend)

Instructions for installing, configuring, and running of the Wireguard server is available here: <https://www.cyberciti.biz/faq/ubuntu-20-04-set-up-wireguard-vpn-server/>

Verify that your Wireguard VPN Server is running by downloading the Wireguard Mobile App and configuring it to connect to the WireGuard VPN Server.   
  
Wireguard Server is not needed to run the Dossier Management System, but recommended in production to secure data traffic between the API Backend and Mobile App users.

### Install and run the Database System

* 1. Run the following docker command to pull the latest MongoDB Image from Docker Hub

docker pull mongo

* 1. Start a mongo instance using the downloaded MongoDB image with the following command.

docker run --name some-mongo -d mongo:tag

*Some-mongo* is the name you want to assign to your container and *tag* is the tag specifying the MongoDB version you want. By default, the authentication is not required to setup, however in a production environment it is important to setup MongoDB admin authentication with the following docker command.

docker run -d --network some-network --name some-mongo \

-e MONGO\_INITDB\_ROOT\_USERNAME=mongoadmin \

-e MONGO\_INITDB\_ROOT\_PASSWORD=secret mongo

### Launch the API Backend System

1. Clone and download the Dossa project file from Github

git clone git@github.com:thachp/dossa.git

1. In the project root directory, create a file .env.local and fill the following variables with your own string.

APPLICATION\_ID=""

MASTER\_KEY=""

JAVASCRIPT\_KEY=""

FILES\_SUBDIRECTORY="../../appdata/incidents"

ENCRYPTIONKEY=""

MONGODB\_URL="mongodb://localhost:27017/dossa "

GRAPHQL\_SERVERURL="http://localhost:1337/graphql"

SERVER\_URL="http://localhost:1337/parse"

VPN\_IPS="::ffff:127.0.0.1"

1. Change into the /api folder and run the following commands to start the API backend.

cd /api

yarn install

yarn start:prod

This will launch the API services at the URL <http://localhost:1337/>

### Launch the Web Application

1. From the root directory, change into the /web directory.
2. Create a file .env.local and fill the following variables with your own string. Note, that the APPLICATION\_ID must match the APPLICATION\_ID in the backend system from the previous instructions. Same with the JAVASCRIPT\_KEY variable.

REACT\_APP\_APPLICATION\_ID=""

REACT\_APP\_JAVASCRIPT\_KEY=""

1. Run the following commands to the start web application.

yarn install

yarn start

This will launch the Web services at the URL <http://localhost:3000>

## Launch the Mobile Application (Optional)

1. From the root directory, change into the /mobile directory
2. Edit the file /common/constants/settings.constant.ts to match those in the .env.local file from the Backend configuration.

export const HOSTNAME = "";

export const PARSE\_API = "https://example.com/parse";

export const PARSE\_GRAPHQL = "http://example.com/graphql";

export const APPLICATION\_ID = "";

export const JAVASCRIPT\_KEY = "";

1. Run the following commands to the mobile application in Expo environment.

yarn start

For more information about EXPO, visit <https://docs.expo.dev/>

### Setup Nginx for Reverse Proxying (Production)

The following installation and configuration instructions are only necessary if Dossa is to be deployed onto a production environment. The operating system used will be Ubuntu Bionic.

1. Update the APT package cache and install the Nginx web server using the package manager:

apt install

apt install nginx

1. Edit the Nginx.conf file to match the following Nginx configuration.

server {

listen 80;

listen [::]:80;

access\_log /var/log/nginx/reverse-access.log;

error\_log /var/log/nginx/reverse-error.log;

location / {

proxy\_pass http://127.0.0.1:3000;

}

}

For instructions on configuration TLS on Nginx, please visit: <http://nginx.org/en/docs/http/configuring_https_servers.html>

## Submit a corruption incident through DOSSA, the Mobile Application

1. *Click the “Log in” button in the top right corner of the app.*

**

1. *If you already have an account, log in with your account name and password. If you need an account, click on the link below that states “Need an account?”*
2. *If you need to create an account, choose a unique username and password. By default, the password requires at least 6 characters. This function could be changed to address new password requirements.*

## *Analyst Dashboard*

### *Create a New Class*

1. *Once logged in, click on the link at the top labeled “Classes”. This will enable you to create a new class of students.*

**

1. *Click on “+ Add Class”.*

*Graphical user interface, text, application

Description automatically generated*

1. *Enter a class name and its description. The class name must be unique.*
2. *Click “Add Class” to add the class, otherwise click “Cancel” or outside of the modal to cancel adding the class.*

Graphical user interface, application

Description automatically generated

## *Institutions List*

1. *To access the reporting feature, from the Schedule module, click on “Generate Report” near the top right of the page.*

*Graphical user interface, website

Description automatically generated*

1. Table

   Description automatically generated*By default, all events are generated and displayed.*