# reconciliation app

## Docker Setup

Use this guide if you want to use Docker in your project.

Built with Docker v18.03.1-ce.

### Getting Started

Update the environment variables in *docker-compose.yml*, and then build the images and spin up the containers:

$ docker-compose up -d --build

By default the app is set to use the production configuration. If you would like to use the development configuration, you can alter the APP\_SETTINGS environment variable:

APP\_SETTINGS="project.server.config.DevelopmentConfig"

Access the application at the address <http://localhost:5002/>

#### Testing

Test without coverage:

$ docker-compose run web python manage.py test

Test with coverage:

$ docker-compose run web python manage.py cov

### Run the app

$ docker-compose run web

## Setup (without Docker)

Use this guide if you do NOT want to use Docker in your project.

### Getting Started

Create and activate a virtual environment, and then install the requirements.

#### Set Environment Variables

Update *project/server/config.py*, and then run:

$ export APP\_NAME="reconciliation"  
$ export APP\_SETTINGS="project.server.config.ProductionConfig"  
$ export FLASK\_DEBUG=0

By default the app is set to use the production configuration. If you would like to use the development configuration, you can alter the APP\_SETTINGS environment variable:

$ export APP\_SETTINGS="project.server.config.DevelopmentConfig"

Using [Pipenv](https://docs.pipenv.org/) or [python-dotenv](https://github.com/theskumar/python-dotenv)? Use the *.env* file to set environment variables:

APP\_NAME="reconciliation"  
APP\_SETTINGS="project.server.config.DevelopmentConfig"  
FLASK\_DEBUG=1

#### Run the Application

$ python manage.py run

Access the application at the address <http://localhost:5000/>

#### Testing

Without coverage:

$ python manage.py test

With coverage:

$ python manage.py cov