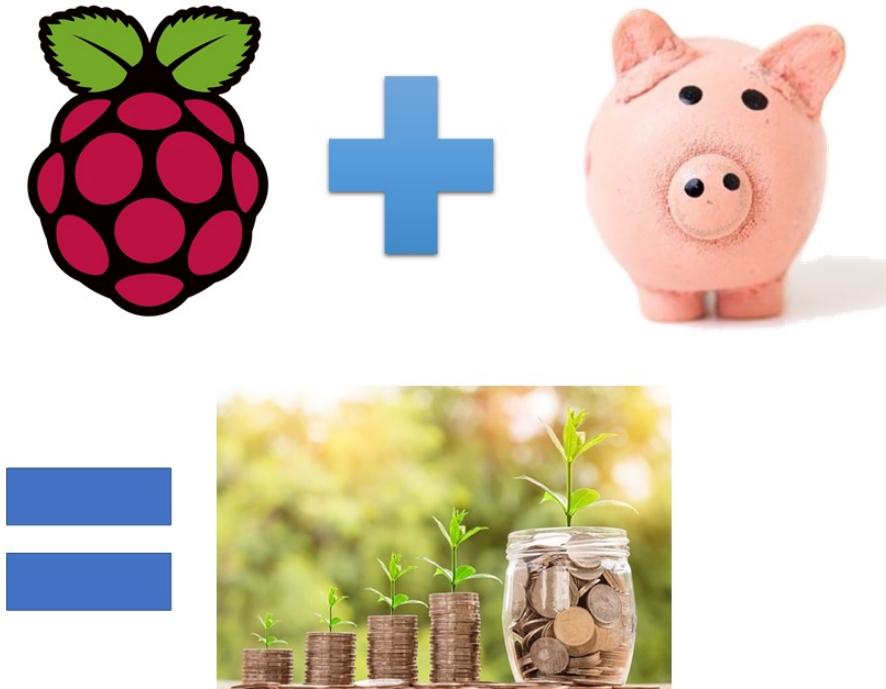


# Manage your Finance with Firefly III and Raspberry PI

Posted on [26th September 2020](#)



Check my RPI articles in [Best Raspberry PI projects article](#) or [peppe8o.com](#) home page. Or subscribe my newsletter (top right in this page) to be notified when new projects are available! Also interested to start 3D printing with a cheap budget? Visit my [cheap 3D printers list](#)

Having finance under control is a good way to correctly manage money and analyze what expenses can be reduced to save. Usually many internet services offer solution to manage your money, but privacy issues raised attention on data saved in cloud, so more and more people look for self hosted solutions able to grant data

reside in their home.

## What is Firefly III

[Firefly III](#), a free and open source personal finance manager, aims to get you aware where your money are spent, so that you can organize better your payments and verify where you should look if you want to make saving.

In this tutorial I'm going to show you how to install FireFly III in a credit card sized, cheap computer: Raspberry PI. I'm going to use Raspberry PI 3 model A+, but this guide applies also to newer Raspberry PI boards.

## What We Need

As usual, I suggest adding from now to your favourite ecommerce shopping chart all needed hardware, so that at the end you will be able to evaluate overall costs and decide if continuing with the project or removing them from shopping chart. So, hardware will be only:

- [Raspberry PI 3 Model A+](#) (including proper [power supply](#) or using a smartphone micro usb charger with at least 3A) or [newer Raspberry PI Board](#)
- [high speed micro SD card](#) (at least 16 GB, at least class 10)

Check hardware prices with following links:

## Step-by-Step Procedure

# Preparing Environment

Start [installing Raspberry PI OS Lite.](#)

Make your operatin system up to date, from command line:

```
sudo apt update -y && sudo apt upgrade -y
```

Then [install Docker and Docker compose](#):

```
curl -fsSL https://get.docker.com -o get-docker.sh  
sudo sh get-docker.sh  
sudo usermod -aG docker pi  
sudo apt install docker-compose
```

Reboot or reload your profile to get docker privileges to pi user.

## Preparing Build and Compose Files

You will need to set your favourite values for some credentials/parameters. It is better to prepare their values from now, so that you will be ready to set them. This tutorial will include following values, so you have to change them with your ones:

- “fireflyiiiDb” (name for database in MariaDB dedicated to your Firefly III)
- “fireflyiiiUser” (user for database access in MariaDB)

- “fireflyiiiPassword” (password for database access in MariaDB)

You will also need an APP\_KEY, which needs to be a random string of 32 characters. This APP\_KEY can be generated from your terminal with command:

```
head /dev/urandom | LANG=C tr -dc 'A-Za-z0-9' | fold -w 32
```

Start preparing the folder where files will be collected. Folder name will be used, by default, from docker compose to name container services. Create and enter it:

```
mkdir fireflyIII  
cd fireflyIII
```

Folder will include following files, which I'm going to show you one by one:

Dockerfile  
mySqlScript.sql  
docker-compose.yml  
.env

## Dockerfile

This file is used to build MariaDB container from a lite debian base image. Please refer to [Create custom MariaDB container](#) article for more detailed info:

```
FROM debian

RUN apt update -y && apt install mariadb-server -y
RUN sed -i 's/bind-address/bind-address = 0.0.0.0 #/i' /etc
RUN mkdir /mysql_scripts
COPY mySqlScript.sql /mysql_scripts/
RUN /etc/init.d/mysql start && mysql -uroot --passw
CMD /etc/init.d/mysql start && tail -f /dev/null
```

## mySqlScript.sql

Also this script refers to MariaDB custom container creation. It differs from original one because the create database command is issued at start. Consider that this script will run only once at first boot during building process:

```
CREATE DATABASE fireflyiiiDb;
create user fireflyiiiUser@'%' identified by 'fireflyiiiPas
grant all privileges on fireflyiiiDb.* to fireflyiiiUser@'%'
flush privileges;
quit
```

## docker-compose.yml

This is a mix between MariaDB compose file and Firefly III file.

```
# My MariaDB
# by peppe80.com
version: '3'
```

```
services:
  fireflyiii:
    image: jc5x/firefly-iii:latest
    volumes:
      - firefly_iii_upload:/var/www/firefly-iii/storage/upload
    env_file: .env
    ports:
      - 80:8080
    depends_on:
      - fireflyiidb
  fireflyiidb:
    build: .
    restart: always
    volumes:
      - firefly_iii_db:/var/lib/mysql

volumes:
  firefly_iii_upload:
  firefly_iii_db:
```

## .env

env file includes all configurations needed by Firefly III service. Get .env file from FireflyIII example file and save it with correct name:

```
wget -O .env https://raw.githubusercontent.com/f...
```

Customize .env file:

```
nano .env
```

Identify and change following parameters:

*APP\_KEY=SomeRandomStringOf32CharsExactly -> change with your random generated APP\_KEY*

*DB\_DATABASE=firefly -> change to  
DB\_DATABASE=fireflyiiiDb*

*DB\_USERNAME=firefly -> change to  
DB\_USERNAME=fireflyiiiUser*

*DB\_PASSWORD=secret\_firefly\_password -> change to  
DB\_PASSWORD=fireflyiiiPassword*

You can edit or leave with default values remaining parameters.

## Run Firefly III

Finally, start service by issuing following terminal command:

```
docker-compose up -d
```

Your Raspberry PI will automatically start from Debian image. It will update the base image and install MariaDB, then using sql script to initialize your database container.

Then, still automatically, it will download all FireFly III packages to initialize related container. Once finished you will have back your terminal prompt. Setup operations

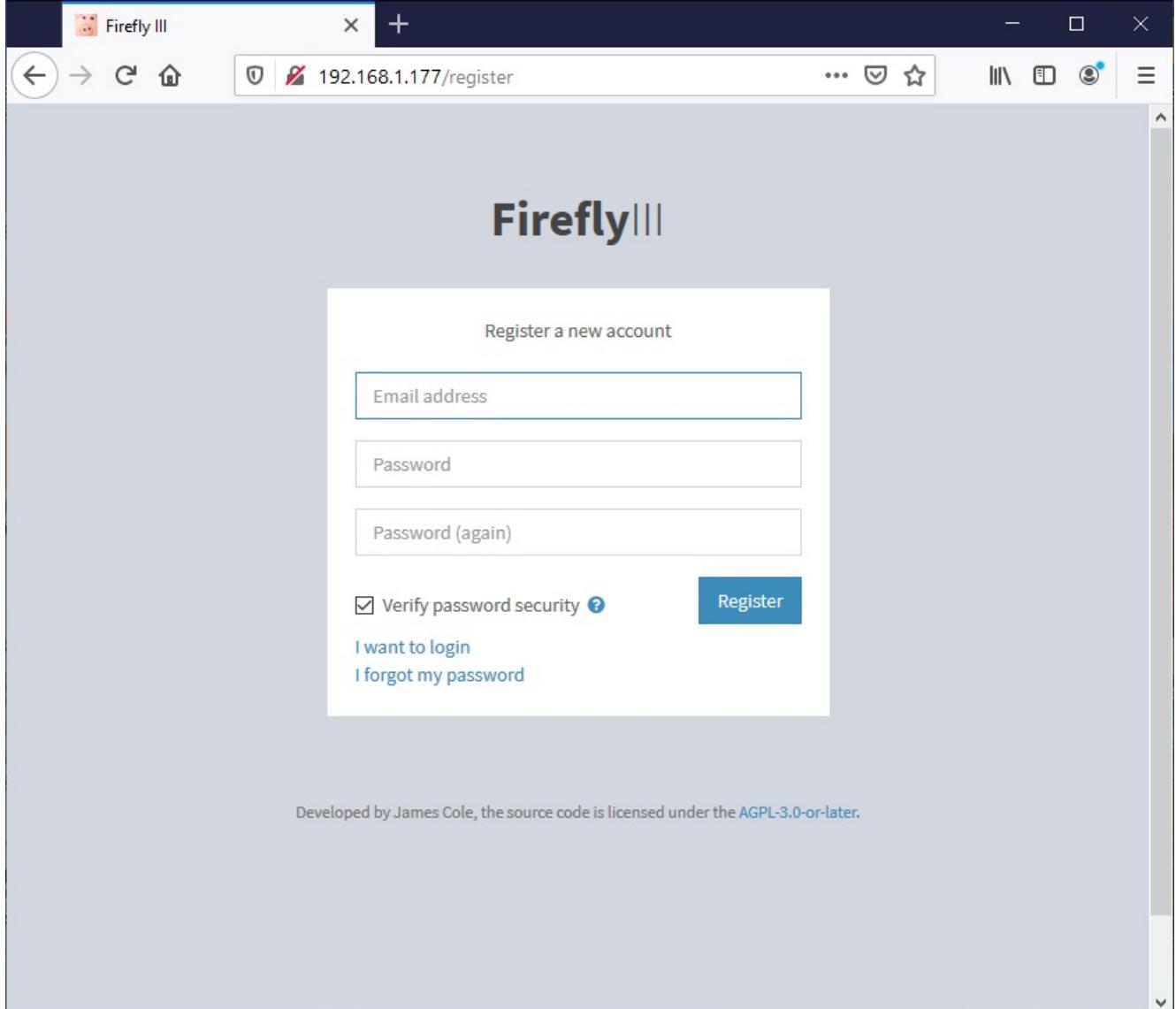
could continue also after containers are created, so check Firefly III web container logs with following command:

```
docker logs fireflyiii_fireflyiii_1
```

Please, note that if you used a different folder name, you will have firefly service named with folder name before underscore character. Operations should finish when this command results in something similar to following one:

```
...
Updated version.
+-----
| |
| Thank you for installing Firefly III, v5.4.2!
| |
| |
+-----  
Go!  
AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 127.0.0.1. You probably need to run apache2 -k reconfigure.  
AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 127.0.0.1.  
[Fri Sep 25 23:22:55.451094 2020] [mpm_prefork:notice] [pid 1] Starting pre-forking daemon.  
[Fri Sep 25 23:22:55.451308 2020] [core:notice]
```

From here your FireflyIII service should be ready. In your favourite web browser, use [Raspberry PI IP address](#) (mine one is 192.168.1.177) for URL. Page will be redirected to Firefly III registration URL (give more minutes in case of connection refused error and try again):



Insert your email address and choose your password (consider that your password must be at least 16 digits, include upper case, symbol, chars and number).

Then click register button. You will be redirected to Firefly III home page:

The screenshot shows the Firefly III web application. The left sidebar contains a search bar and links to various sections: Dashboard, FINANCIAL CONTROL (Budgets, Bills, Piggy banks), ACCOUNTING (Transactions, Automation), OTHERS (Accounts, Classification, Reports, Export data). The main content area is titled 'Welcome to Firefly III!' and shows the 'Getting started' page. It includes instructions for entering bank information (Bank name, Balance) and savings account details (Savings balance). It also provides options for language selection (Language: English (US)).

You are now ready to use your Firefly III service!

For more info and user guide, please refer to [Firefly III official documentation page](#).

## Final Operations

At procedure end, it is a good practice entering MariaDB container to change root password for security.

Enjoy!