Installation Manual

Prepared by: Bikram Gill, 5938923

Wesley Nantais, 5903455

Tyler Applin, 5920616

Brock Jones, 5859723

Eduardo Galindo, 5480504

Emmanuel Guarino, 6038970

Ege Can Kaya, 6150817

4F00 CMS Analysis & Design

April 25th, 2020

To setup this project download a project into a folder, cd into that folder, run sh setup.sh and then edit the config/config.php file to your setup. You will need to change the config/config.php to edit it so the home-file-path reflects the path to reach the root folder. For example if the project is in '/home/student/c4f00g05/public_html/people/Brock then the config/config.php file should have
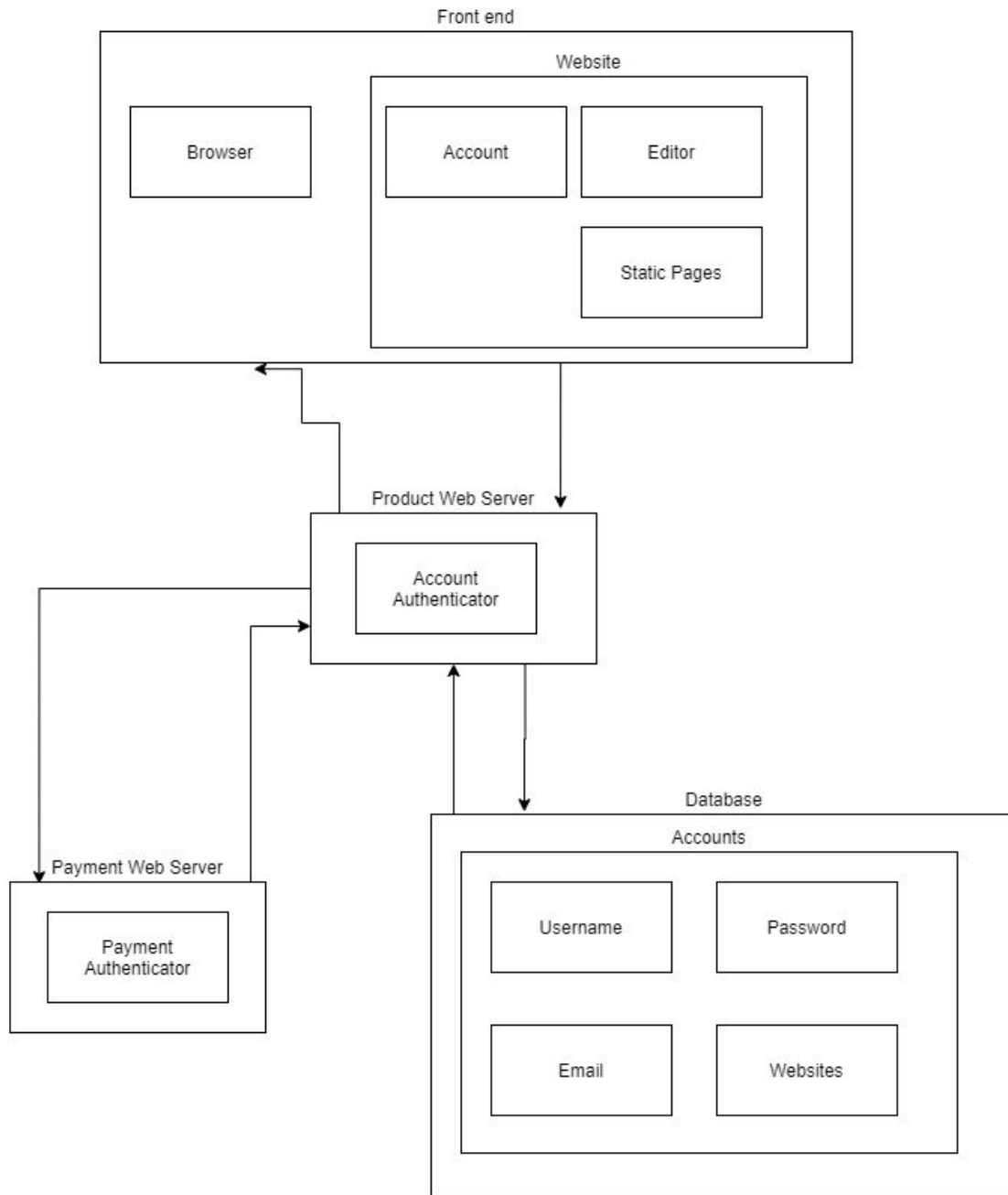`'home-file-path' => '/~c4f00g05/people/Brock'`

To set up database load
https://www.cosc.brocku.ca/~c4f00g05/people/Brock/controller/controller.php?COMMAND=UNIQUE

If you are getting session errors when loading the webpage try deleting your cookies for this site or opening a website in a private browser.

Overall System
UML Architecture Diagram

Front end

Website

Browser

Account

Editor

Static Pages

Product Web Server

Account
Authenticator

Payment Web Server

Payment
Authenticator

Database

Accounts

Username

Password

Email

Websites

This CMS is built using a Model-View-Controller pattern. All of the webpages that are used for displaying the CMS to the client are located in the view folder. They are all PHP files and are loaded by the web server whenever you enter their url. These View webpages have access to the model in order to load data for displaying, although the views never directly update the model only use it for displaying data. To update the model the views send data to the controller using a post request, the controller then makes a call to a model method in order to update the model, and then the controller redirects to whichever view webpage it needs to redirect to. Sometimes the controller needs to send a temporary message to the view (e.g. in case something went wrong) and to send a temporary message to the view the controller sets a session variable containing the message. The model stores all of its data in a postgresql database, the way the model communicates with the database is through using the redbean ORM. The editor stores all of its web pages (web pages generated by the CMS) in a single large JSON string.