# Digital library system in flask

Lukas Prokop, Andi

January 4, 2014

## Contents

## 1 Technology involved

**Python** Our programming language in charge (version 2.7 is used).

**Flask** Flask is a python micro-webframework written by Armin Ronacher. It features a very simple and basic API to provide web content easily and is extensible by plugins.

**SQLAlchemy** SQLAlchemy is used as database binding (object relational mapped) to a postgresql database.

**PostgreSQL** We have decided in favor of postgresql as our database system.

**Jinja2** This is our template engine (equivalently its template language).

**Elastic search** Provides search capabilities to our dataset.

## 2 How to set it up

1. Run `virtualenv .` and `. bin/activate` as described at Flask's installation page.
2. Run `pip install Flask` and `pip install elasticsearch` to install those packages for this repository locally.
3. Run `python index.py` in the project's root directory to start the web application.

4. Visit `http://localhost:5000/`.

## 3 Database layout

We only use two tables. One table is called *documents* which registers all documents. Attributes are:

**id** A basic integer primary key.

**type** Which kind of document this is (e.g. "doc", "attach", "comment")

**title** Title of the document

**author** Who created this document?

**timestamp** When was this document created?

Our second table stores various metadata, which attributes also its name *metadata*.

**document** Which document is this metadata associated with?

**key** An arbitrary key like `pdf.author` for the author in the metadata field of the PDF file.

**value** The corresponding value for the given `key` of `document`.

## 4 How to use it

The requirements state that browsing, searching, inserting and presenting documents must be possible. This corresponds to the following pages of the web application.

**Browsing** Documents can be browsed in the listing. You have to visit the list page.

**Searching** Main (start) page. You can enter various search queries to retrieve documents from the dataset. The set of possible search queries is listed at the syntax page.

**Inserting** The insertion can be done by visiting the upload page.

**Presenting** Each document (and its associated documents) are presented on a separate page.