

Project trckr

Technical Article

Ankeshian, Gabriel

`ankesgab@students.zhaw.ch`

Balidis, Dimitri

`baliddim@students.zhaw.ch`

Christen, Luca

`chrisluc@students.zhaw.ch`

Jossi, Savino

`jossisav@students.zhaw.ch`

Milenkovic, Daniel

`milendan@students.zhaw.ch`

Nominato, Angelica Helena Moreira Alves

`moreiane@students.zhaw.ch`

Pacassi, David

`pacasdav@students.zhaw.ch`

May 2018

The main goal of the present article is to describe the idea, goals and main functionalities of the web based application trckr. We developed trckr for everyone who works on a project and needs an intuitive and simple web tool to accurately track their time spent on different tasks.

The backend is written in Python with the help of the web application framework Django and the frontend with the Javascript UI framework Vue.js. Both technologies were new to most team members, but have proven themselves effective and learning them were ultimately a benefit the outcome of the project.

In order to compete with similar tools and web services, trckr focuses on performance and usability. To distinguish trckr from the competition, many features are planned to manage projects and tasks in a user-friendly manner. This will allow the user to leverage trckr to handle the ever increasing complexity in project management and task tracking found in large companies. Despite being targeted at large companies, trckr will remain open source and anybody can contribute, covering cases we might have never dreamed of.

Contents

1	Introduction	4
1.1	Objectives	4
1.2	Main Features	4
2	Technologies	4
2.1	Django	4
2.2	PostgreSQL	5
2.3	Vue.js	5
3	Results	5
3.1	API	6
3.1.1	Endpoints	6
3.2	User Interface	6
4	Outlook	9
5	Conclusion	9
6	Bibliography	9

1 Introduction

Time and task tracking are important activities in many businesses to gain insights on the productivity of a team, requiring appropriate tools allowing easier and more accurate time tracking. Many processes and methods have been developed in the past to cover this need. Unfortunately most of them address just a certain need or have been fine tuned to a specific company or team. This necessarily leads to a loss of experience that could have been leveraged by other teams but is also not generic enough to adapt to different processes, causing other companies to inappropriately adapt to the tool instead of the tool adapting to the company.

1.1 Objectives

The goal with trckr is to develop and distribute a time tracking web application that is easy to understand and use. The most important requirement is to require just a few steps to track all the required information. Only this will keep the user engaged and raise the accuracy of the provided data.

1.2 Main Features

The user is able to:

- register and login to trckr
- create and edit projects
- create, track and edit tasks
- visit trckr on any device

2 Technologies

2.1 Django

Django is an open source web framework written in Python. It encourages fast, clean and simple development of web applications. One of the main advantages is it's fast setup, enabling the developer to create applications swiftly through it's Model-View-Presenter scheme. Django also comes with support for various databases. Many users compare Django to Ruby

On Rails but written in Python. Django also follows the DRY principle (Don't Repeat Yourself).

2.2 PostgreSQL

A PostgreSQL database stores all the data for trckr. PostgreSQL is fully supported by Django and configuration and management is minimal, because the database access is handled by Django.

2.3 Vue.js

Vue.js is a progressive framework for building user interfaces.[1] Vue.js is used for trckr mainly for its simplicity and the rather shallow learning curve it provides to unexperienced developers. The features that Vue.js provides allow the creation of data structures that can easily be displayed in on a website. This and the ability to easily make calls to the backend make it a good fit for trckr.

3 Results

The architecture was kept fairly simple, as illustrated in figure 1, using Django with PostgreSQL in the backend and a frontend based around Vue.js.

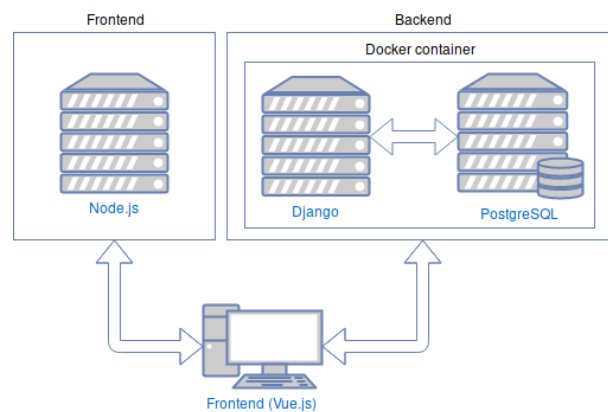


Figure 1: Architecture of trckr

3.1 API

The backend of the trckr application implements a RESTful API using the Django REST framework. The API provides basic CRUD operations for all the entities available in the database. There are five main endpoints to retrieve and save data on the server: authentication, user, projects, tasks and time entries. Except for the authentication and user endpoints, all endpoints need an authentication token to be accessed.

3.1.1 Endpoints

authentication allows users to retrieve an authentication token from the server to access the other parts of the API. Via this endpoint, one can also invalidate the token.

user is only used to create new user accounts.

projects lets user create, read, update and delete projects as well as display all the tasks associated with a project.

task used to create, read and update tasks for a given project. There also exists a way to list all relevant time entries for a task.

time entries also used for create, read, update and delete operations

Each object of an entity has a unique ID. This ID can be used to retrieve information for that specific object by providing it in the URL when calling the server. This is necessary when updating an object via a POST request.

3.2 User Interface

When opening the trckr website the first time you are presented with the login screen as shown in figure 2. For users who have not yet registered an account, this can be done here by entering some basic information like username, password, email address and first and last name (Figure 3).

Once the registration process has been completed, the user can log in at the login page. Once the user is logged in, the navigation bar at the top of the interface will contain links to the dashboard, the projects page and the time entries page as well as a logout button. The dashboard displays graphs... TODO

trckr x

← → ↻ 🏠 Sicher | <https://trckr.tvlr.ch/login>

trckr Login Register

Login

Username

Password

Login Don't have an account yet? [Register](#) now!

Figure 2: The trckr login page.

trckr x

← → ↻ 🏠 Sicher | <https://trckr.tvlr.ch/register>

trckr Login Register

User registration

Username

Password

Email

First name

Last name

Register Already have an account? [Login!](#)

Figure 3: The trckr registration page.

The projects page shows a table of all projects that the currently logged in user is a part of, seen in figure 5. There is a search box above the projects list that allows a user to filter for a project or a group of projects containing the given keywords. To create a new project the user will have to navigate to the projects page and click on the "Create project" link which will open the project creation form. The form asks for the name of the project and optionally allows the user to enter an description.

Each project can be viewed in more detail when the project inside the table is clicked, as shown in figure 5. The project page contains the name, the description and a table of all the tasks in the selected project.

Clicking on a task will display the details of said task, at the current state that is only the name and the description of the task.

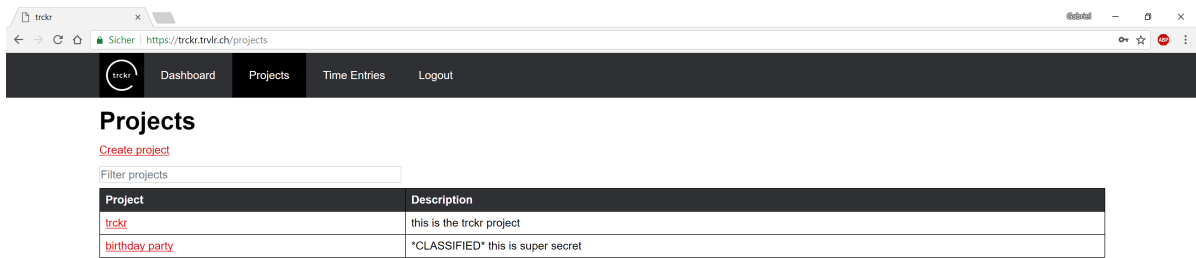


Figure 4: The projects page.

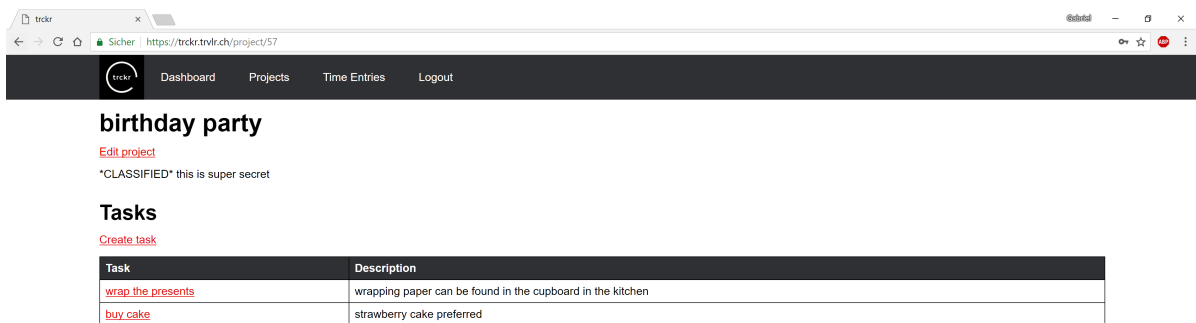


Figure 5: The projects page with the table of all projects.

On the "Time Entries" page a user can create a time entry for a specific task of a project. There is a link on the page that leads to the time entry form. In said form the user has to choose a project first and then a task of the project for which the time entry should be created.

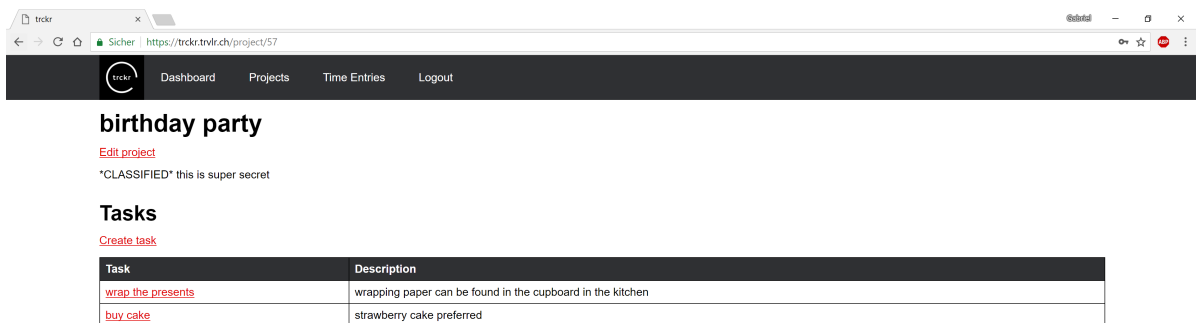


Figure 6: The form to add a time entry to a task.

All time entries will be displayed on the time entry page.

4 Outlook

Outlook...

5 Conclusion

Conclusion...

6 Bibliography

Bibliography...

References

[1] <https://vuejs.org/> ; as 10.05.2018

List of Figures

1	Architecture of trckr	5
2	The trckr login page.	7
3	The trckr registration page.	7
4	The projects page.	8
5	The projects page with the table of all projects.	8
6	The form to add a time entry to a task.	8