Kalin Karaliev

Location:	Kingston upon Thames, KT2, London, UK
Mobile:	07860530178
Email:	kalin.karaliev@gmail.com

Profile

As an IT graduate with First-Class Honours from the University of West London, having recently completed a 18-month Python Web Developer program with top results and with strong and steady interest in Web Development, Coding Languages, Computer Networking and Data Science, I am now seeking a role in an organisation which will give me the opportunity to apply and expand my knowledge in real working environment.

My core technical skills include Python, Django, JavaScript, Express.js, React, HTML/CSS, SQL, Linux etc.

Previously I have worked as a Health and Safety Officer, Site Person in Charge and Supervisor on a number of different projects for London Underground as well as a Graphic Designer in a daily newspaper. The experience of working in dynamic, diverse environments, the ability to solve problems and communicate with various stakeholders can be transferred across and utilised for the benefit of your organisation.

Please check my portfolio web page and GitHub account.

Web Dev Projects

✔ Artist's Web Portfolio – WordPress

The raison d'être of this WordPress website is to create an online presence of an artist, my father, by presenting a comprehensive portfolio of his work. I take some pride in the fact that it is a solo effort of mine - from physically taking the photos, to editing them with Gimp, to arranging the domain name and hosting, to the actual design, implementation and deployment. It was part of the my final year project at University of West London which has been awarded "Best Project on the BSc Information Technology course".

✔ Vinylarium – React + Node.js + Express + PostgreSQL

Vinylarium is a mock online shop for vinyls. (Please note, the back end server is hosted under Render's free tier, hence it needs approx. 30 seconds to spin up and send the initial response to the front end app.) Users can register, log in (including with Google credentials), browse the catalogue, manipulate their cart and place orders after simulated payment. Apart from this users with admin privileges (the back end confirms/rejects the requests depending on the submitted JWT) have access to admin interface, allowing them to perform CRUD operations on the DB, e.g. adding new items to the catalogue, editing or deleting users etc.

The back end of the shop was created with Express JS. The app connects to a Postgres DB (hosted at ElephantSQL) and provides more endpoints than the front end currently utilizes – my intention is to keep expanding and improving the app. The DB design uses intermediary tables to avoid many-to-many relations between some entities, e.g. album/order. The ERD can be seen here.

Dedicated middlewares are used to verify users' authentication credentials and admin privileges, while passwords are stored in encrypted format. OAuth 2.0 via Google is also implemented.

✔ Phonotheque – Python + Django

Phonotheque is a web application combining social media, online forum, data scraping and data storage features. Its objective is to act as a platform where users can share their favourite music albums and communicate with one another. The back end has been created using the Python-based framework Django. The About section provides in-depth info about the technicalities while the app itself can be tested safely and effortlessly with mock registration credentials – no personal data, including an actual email address, needs to be provided in the process.

The app also has admin moderation interface with different levels of authorization implemented.

Identification of an album which is to be added to the DB is achieved by fetching information from Wikipedia - *not the API but the actual web page*. Of course this approach is more error- and omission-prone compared to an actual API search and, of course, using a spacialized music API (e.g. Spotify’s) would always be the natural choice but in my opinion the project as it is represents a good effort in data scraping.

Eventually I will go for the practical solution but I found the experiment of building an app capable of searching actual webpages and getting relevant information from them it quite interesting and beneficial.

It is an ongoing project which I am really enthusiastic about – I am a music aficionado myself and believe that the idea has its merits. The potential for expanding it by adding new features and components, improving its design, security and functionalities is virtually unlimited.

✔ Spotify Playlist Creator – React

Spotify Playlist Creator is an app created with React which allows users to search Spotify's API, create a playlist by adding and removing tracks, give it a name and eventually save it to their actual Spotify account. It utilizes Implicit Grant Flow meaning that the process of logging in / obtaining JWT is carried out entirely on the client side. The app will ask the user to log in and grant access to their account (scope is limited to 'playlist-modify-private playlist-modify-public'). If approved Spotify will respond with an URL containing a hash from which an access token and related data can be derived.

✔ Mini Reddit – React

Mini Reddit was created with React and react-router-dom and is extracting data from Reddit's API. The index page loads, filters and displays data (properties like "accounts_active", "subscribers", "created_utc", "public_description", "icon_img") about 19 subreddits I am following. (This initial loading process due to its nature takes a few seconds, I am afraid). The user can then select a subreddit (or ALL) and sort the posts by different criteria ("Best", "Top", "Hot", "Controversial"). If a post preview <div/> is clicked a details page will be displayed, including the related user comments. Again, data will be filtered and truncated before being processed for rendering as the original json files received from the API are immense. The app offers search functionality as well, either within a specific subreddit or for all subreddits.

✔ QuiQui (Quick Quiz) – React

QuiQui (Quick Quiz) is a web app build with React JS and utilizing the Open Trivia Database API. The user can request and answer a 10-question quiz by customizing its topic, difficulty and type of questions (boolean or multiple choice). Once the questions have been answered (or have been left unanswered), the app will verify the answers and count the correct ones.

Skills

- ✔ Programming Languages: Python, JS, HTML/CSS, PHP, SQL.
- ✔ Experience with Django, Express.js, React, WordPress, jQuery, testing frameworks, Adobe Creative Cloud.
- ✔ Deep understanding of computer networks.

- ✔ Linux and Windows user.
- ✔ Adaptable, committed, able to deliver under pressure and in tight timescales.
- ✔ Energetic, reliable and friendly with great communication skills.
- ✔ Clean driving license.

UK Employment History

➔ **Health and Safety Officer / Site Manager / Site Person in Charge**
@ Delatim, London
(May 2021 – present)

Main tasks performed:

Providing protection for Delatim workers when on London Underground infrastructure. Responsible for all aspects of health and safety on shift, including ensuring workers are not endangered by moving trains, traction current and other potential hazards.

Main responsibilities:

- ✔ Supervision of teams consisting of a varying number of colleagues.
- ✔ Producing detailed reports using Excel spreadsheets
- ✔ Taking decisions when unexpected circumstances arise
- ✔ Undertaking site inspections to ensure work is completed to expected quality standards
- ✔ Working to very tight timelines.

➔ **Protecting Workers on the Track - Engineering Hours**
@ Cleshar, London
(February 2020 – May 2021)

Main tasks performed / Main responsibilities:

(Please refer to the most recent job description)

➔ **PWT, Site Person in Charge and Supervisor**
@ AGS, London Office
(March 2010 – February 2020)

Main tasks performed / Main responsibilities:

(Please refer to the most recent job description)

Education

➔ **University of West London BSc (Hons) Information Technology**
(September 2017 – June 2020)

Related Modules Covered:

- Computer Architecture
- Java Programming
- HTML & CSS
- SQL
- JavaScript & PHP

- Networks & Security
- Web & Mobile Application Security
- Databases & Analytics
- Final Year Project (won award for Best Project on the BSc Information Technology course.)

➔ **SoftUni – the Software University**
(February 2021 – August 2022)

Modules Covered:

- Programming Fundamentals
- Python Advanced
- Python Object Oriented Programming
- Python Web Framework (Django)
- JS Advanced
- JS Applications
- Computer Networking Advanced

Hobbies & Interests

- ✔ Sports - skiing, kitesurfing
- ✔ Music – jazz, blues, alternative rock, extreme metal… and everything in between
- ✔ Arts - music, cinema, painting
- ✔ Travelling and exploring new places and cultures