AI-SYSTEMS.TODAY

Leverage human potential

KYRIACOS ANTONIADES

MOTIVATION

AI-SYSTEMS.TODAY
LEARNING WITH DEEP LEARNING

www.ai-systems.today

MOTIVATION

"*Integration, integration, integration*"

Nowadays, looking to further development of career, and professional prospects by applying for the candidate position of Python Developer with **EPO** at **The Hague**.

Fascinated with **EPO**'s mission, vision and values providing services, quality management and consultation on patent applications. Impressed with **Espacenet**'s free worldwide collections of patent publications of over 100 million documents from over 100 countries. The cross browser-device compatible extension enables to search, find, and innovate on the improved features of the EPO's public database. It allows for keyword search (authors/technologies) with parameters (e.g. date ranges, country codes), filtering, classifications, as well as it incorporates intelligent algorithms to rank and refine search results of documents accordingly. Documents thence can also be translated in many other languages. It offers advanced search capabilities for more power and precision, wherein supported with pop-up tips.

Genuinely excited to work on the frontiers on migrating available legacy scripts as well as to customize utilized search components and control arrays of widgets to develop the next generation of JupyterHub/Lab extensions.

The extensions allow users to perform more improved searches, retrieve documents, activities, as well as effectively administering their intercommunications. They generate function arguments on the Model View (MV) paradigm by importing interact on the Ipywidgets python GUI package. Extensions intercommunications management is achieved with JupyterHub/Lab's REST API configuration of components, such as, to provision newly produced research, also, to save, edit, and update synonyms and annotations. Project and knowledge management of the control extensions are carried out in the JIRA platform, integrating CI/CD services (GitHub/Bitbucket/Jenkins) with Jupyter notebook instances to create/share live python code, equations, visualizations, and widgets in real-time.

Independent, motivated, and reliable, has an aptitude to challenge, change and impact for the **EPO**'s achievement of project's aim to enable most simplified access to the most relevant systems at the **EPO**. This requires solutions to the following challenges:

1. Select the right JupyterHub/Lab platform (local, virtual, cluster) according to user capacity requirements.
2. Based on the platform requirements, setup Kubernetes Cluster(s) to install JupyterHub/Lab server(s).
3. Customize JupyterHub/Lab environment by switching ad hoc between the hub and lab mode.
4. Customize JupyterHub/Lab Kernel by selecting ad hoc Jupyter Stacks (DataScience, TensorFlow, PySpark) Containers or build necessary python frameworks and ecosphere, by writing own Docker files.

5. Support DevOps, to implement search extensions, installing Ipywidgets (python GUI package) on the Kernel by programming on the external API.
6. Enable extensions' cross-device compatibility and themes using React.js.
7. Integrate JupyterHub/Lab with version control management system (GitHub) using Gitpuller.
8. Improve CI/CD pipeline by integrating JIRA platform for project/knowledge management, supporting GitHub as well as Bitbucket/Jenkins.
9. Administer domain security (https, DNS), authentication (OAuth2, GitHub, Google) for administrators and users, and load balancing (Kubernetes auto-scaling) to manage traffic between servers and authenticated users.
10. Configure servers/users using JupyterHub/Lab internal REST API (components) to manage extensions interconnections (GUI administration).
11. Estimate deployment costs for each of the platforms
12. Provide API documentation (user/technical) on the extensions, using Sphinx/docutils.

To provide change, as an Infrastructure Engineer, keen to be involved with the Container as a Service (CaaS) creation/tearing of Kubernetes (kubectl) and JupyterHub/Lab servers (Helm charts). Can set up, customize, and administer a JupyterHub/Lab server on any cloud platform within hours. Maintain, improve, and administer architecture efficiency, and integrate CI /CD python kernel/modules within days to produce the cloud development platforms as and when required (config.yaml file). With attention to deployment costs.

As a programmer in the python framework/ecosphere using CaaS development platform, eager to build state-of-the-art JupyterLab extensions to improve speed and ease of use of search GUI widgets and find ways to add the further advanced search functionality. Excited to configure platform using REST API components to enable extensions' administration between authenticated users and servers (jupyterhub_config.py). More than happy to produce API documentation (user/technical) of the extensions developed, using Sphinx/docutils.

Motivated by a desire to make an impact, will seek ways to integrate machine learning/deep learning (ML/DL) methodologies to provide for unprecedented personalized search experiences that dynamically adjust to individual users' needs and requirements.

Determined to contribute with own experience and knowledge to leverage **EPO** corporate GUI components to extend support of the Ipywidgets library development roadmap.