Architecture overview

This section contains the following topics:

Architecture overview

The Crosswork Workflow Manager architecture is a microservice-based solution that operates on top of the Kubernetes container orchestration system. This section shows a diagram presenting its core architectural components along with short descriptions of each.

Figure 1: Architecture overview
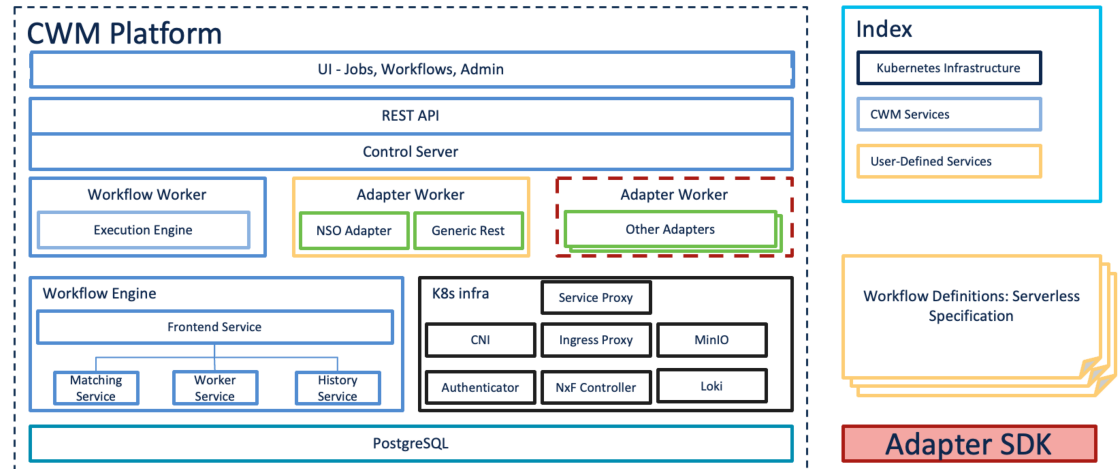
- **User Interface (UI)**: allows operators to add and instantiate workflows, enter workflow data, list running workflows, monitor job progress. The **Admin** section of the UI enables adding workers, managing worker processes and assigning activities from adapters to workers.

- **REST API**: includes all interaction with the CWM application: deploying adapters, publishing and instantiating workflows, managing workers, resources and secrets.

- **Control Server**: dispatches API requests to relevant microservices.

- **Workflow Engine**: it is the core component that conducts how workflows are handled; it interprets and manages the execution of workflow definitions.

- **Execution Engine (Workflow Worker)**: it is responsible for executing the workflow tasks. It receives the workflow tasks from the **Workflow Engine**, executes them in the correct order, and sends the results back to the **Workflow Engine**.

- **Adapter Workers**: they are processes responsible for executing the tasks defined in workflow definitions and adapter code. They receive the tasks from the **Workflow Worker**, execute them, and send the results back to the **Workflow Worker**. The Execution Workers are capable to load additional adapters as plugins, which allows them to work with different systems and technologies.

- **Adapters**: they interface and integrate with external systems, applications and technologies. Inside them, activities that can be consumed in a workflow are defined.

- **Adapter SDK**: a Software Development Kit that helps developers create new adapters to integrate with external systems.

- **Workflow Definitions**: workflow code written in the JSON format based on the Serverless Workflow specification.

- **K8s Infrastructure**: runtime platform for the CWM application. It is a collection of services that provide the necessary infrastructure to support the deployment and management of the application within a Kubernetes cluster.

- **PostgreSQL**: it is the database used by the system to store and manage its data.