FLOWx.AI

PLATFORM OVERVIEW / flowx-architecture

Contents

PLATFORM OVERVIEW / FLOWX.AI architecture

Let's go through the main components of the FLOWX.AI platform:

- **FLOWX.AI SDKs** - used in the Web (Angular), iOS, and Android applications to render the process screens and orchestrate the custom components
- **FLOWX.AI Designer** - is a collaborative, no-code, web-based application development environment that enables users to create web and mobile applications without having to know how to code:
  - Develop processes based on BPMN 2.0
  - Configure user interfaces for the processes for both generated and custom screens
  - Define business rules and validations via DMN files or via the MVEL, or other supported scripting languages
  - Create integration connectors in a visual manner
  - Create data models for your applications
  - Adding new capabilities by using plugins
 - Manage users access

Microservices

- **FLOWX.AI Engine** - is the core of the platform. It runs the business processes, coordinating integrations and the UI

- **FLOWX.AI Admin** - used to store/edit process definitions (FLOWX.AI Admin Microservice connects to the same Postgres / Oracle database as the FLOWX.AI Engine)

- **FLOWX.AI Scheduler** (part of the core components) - used to store/edit process definitions

- **FLOWX.AI Content Management** (part of the core components) - can be quickly deployed on the chosen infrastructure, preloaded with the needed taxonomies or contents, and then connected to the FLOWX Engine through Kafka events

- **FLOWX.AI License Manager** (part of the core components) - is used for displaying reports regarding the usage of the platform in the FLOWX.AI Designer

- **FLOWX.AI Plugins** - the platform comes with some ready-made integrations, such as a [document management] solution, a plugin for sending various types of notifications, an OCR plugin, and a task management plugin
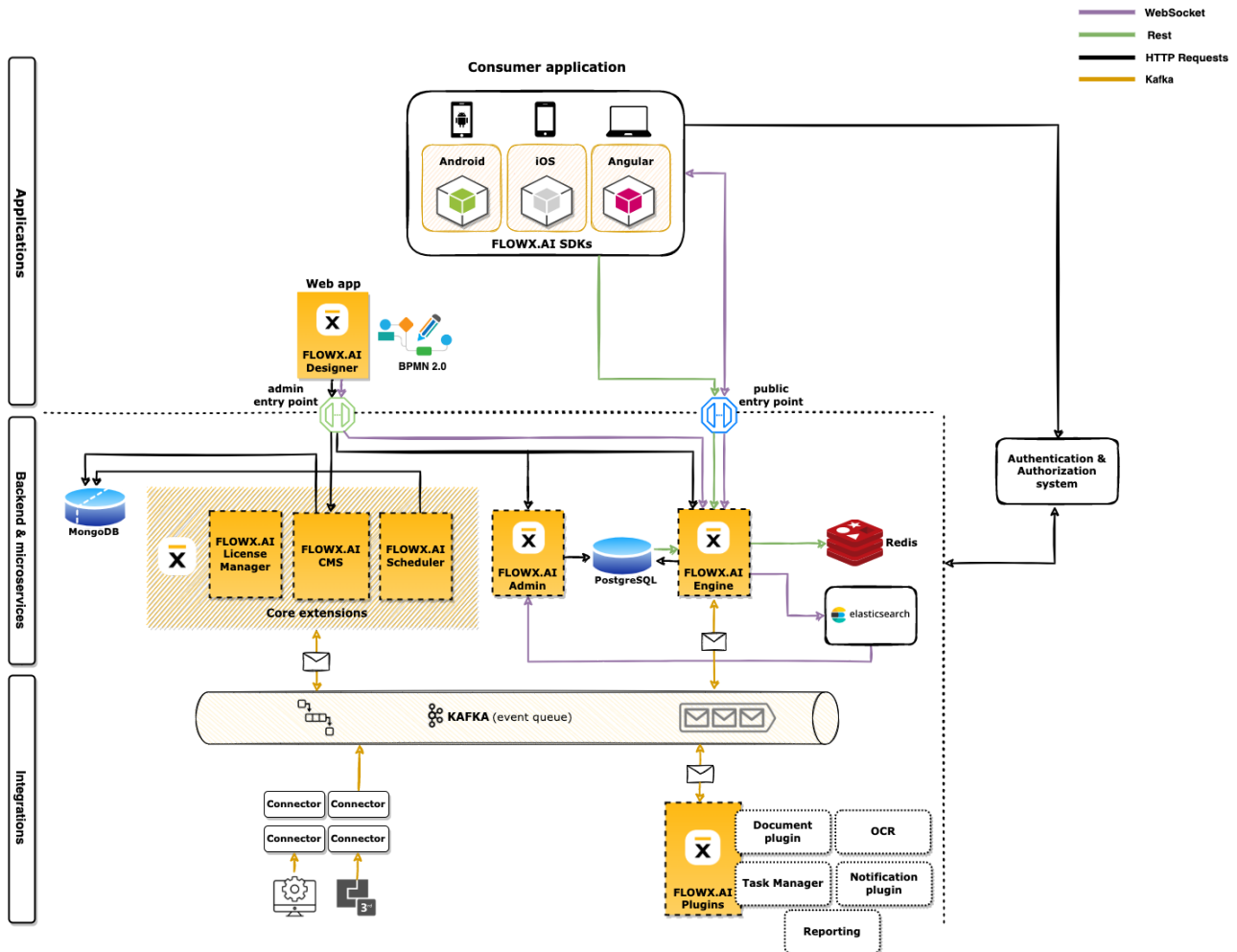
FLOWX.AI Engine

We call it the engine because it's a nice analogy, once deployed on an existing stack, FLOWX.AI becomes the core of your digital operating model.
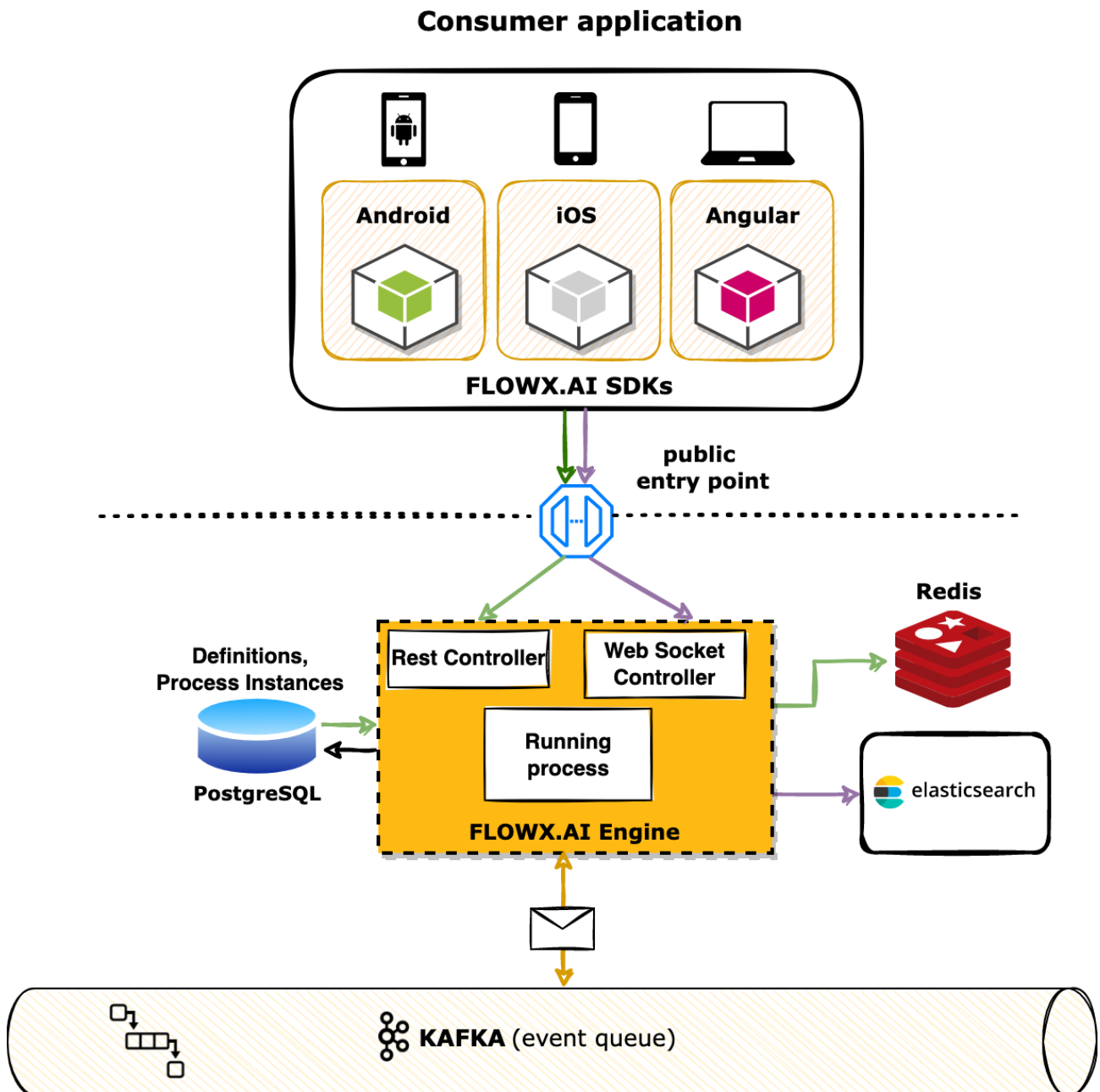
You can use FLOWX Engine to do the following:

- create any type of external or internal facing application
- redesign business processes from analog, paper-based ones to fully digital
  and automated processes

- manage integrations, so you can hook it up to existing CRMs, ERPs, KYC, transaction data and many more

- to read process definitions (if it is connected to the same DB as FLOWX.AI Admin)

FLOWX Engine runs the business processes, coordinating integrations and the omnichannel UI. It is a Kafka-based event-driven platform, that is able to orchestrate, generate and integrate with any type of legacy system, without expensive or risky upgrades.

This is extremely important because often, digital apps used by a bank's clients, for example, are limited by the load imposed by the core banking system. And the customers see blocked screens and endlessly spinning flywheels. FLOWX.AI buffers this load, offering a 0.2s response time, thus the customer never has to wait for data to load.

> » FLOWX Engine

FLOWX.AI Designer

FLOWX.AI Designer is built to administrate everything in FLOWX.AI. It is a web application that runs in the browser, meaning that it resides out of a FLOWX deployment.

The platform has **no-code/full-code capabilities**, meaning applications can be developed in a visual way, available for anyone with a powerful business idea. So we're talking about business analysts, product managers - people without advanced programming skills, and also experienced developers.

The process visual designer works on BPMN 2.0 standard - meaning that the learning curve for business analysts or product managers is quite fast. Thus, creating new applications (e.g. onboarding an SME client for banks) or adding new functionality (allow personal data changes in an app) takes only 10 days, instead of 6 to 8 months.

However, we do support custom CSS or custom screens. Because we're aware each brand is different and each has its own CI, so you need to have the ability to create UIs that respect your brand guidelines.

» FLOWX.AI Designer

FLOWX.AI SDKs

Also, we provide web and native mobile SDKs, so that every app you create is automatically an omnichannel one: it can be displayed in a browser, embedded in an internet banking interface, or in a mobile banking app. Or even deployed as a standalone app in Google Play or AppStore.

Unlike other no-code/full-code platforms which provide templates or building blocks for the UI, ours is generated on the fly, as a business analyst creates the process and the data points. This feature reduces the need to use UX/UI expertise, the UI being generated respecting state-of-the-art UI frameworks.

» Renderer SDKs

FLOWX.AI Content management

This is another Java microservice that enables you to store and manage content. **The go-to place for all taxonomies.** The extension offers a convenient way of managing various content pieces such as lists or content translations. Anything that is under content management is managed by the CMS backend service. To store content, the service will use a MongoDB database (unstructured database). For example, each time you edit an enumeration, the FLOWX.AI Designer will send an HTTP request to the microservice.

» Content Management

FLOWX.AI Scheduler

If you need to **set a timer on** a process that needs to end after X days, you can use the FLOWX.AI Scheduler microservice. It is a service that is able to receive requests (like a reminder application) to remind you in X amount of time to do something.

> ⓘ **INFO**
>
> When you start a process, the process must have an expiry date.

Scheduler microservice communicates with the FLOWX.AI Engine through Kafka Event Queue ↠ it creates a new message (write some data) then will send that message to Kafka (with the scheduler address) → when the reminder time comes up, the scheduler will put back a new message in the Kafka layer with engine's destination (time + ID of the process).

» Scheduler

Authorization & session manager

We recommend Keycloak, a component that allows you to create users and store credentials. It can be also used for authorization - defining groups, and assigning roles to users.

Every communication that comes from a consumer application, goes through a public entry point (API Gateway). To communicate with this component, the consumer application tries to start a process and the public entry point will check for authentication (Keycloak will send you a token) and the entry point validates it.

» Keycloak Documentation

Plugins

Plugins are bits of functionality that allow you to expand the functionality of the platform - for example, we have the following custom plugins:

- FLOWX Notifications Plugin
- FLOWX Documents Plugin
- FLOWX OCR Plugin
- FLOWX Task Management Plugin
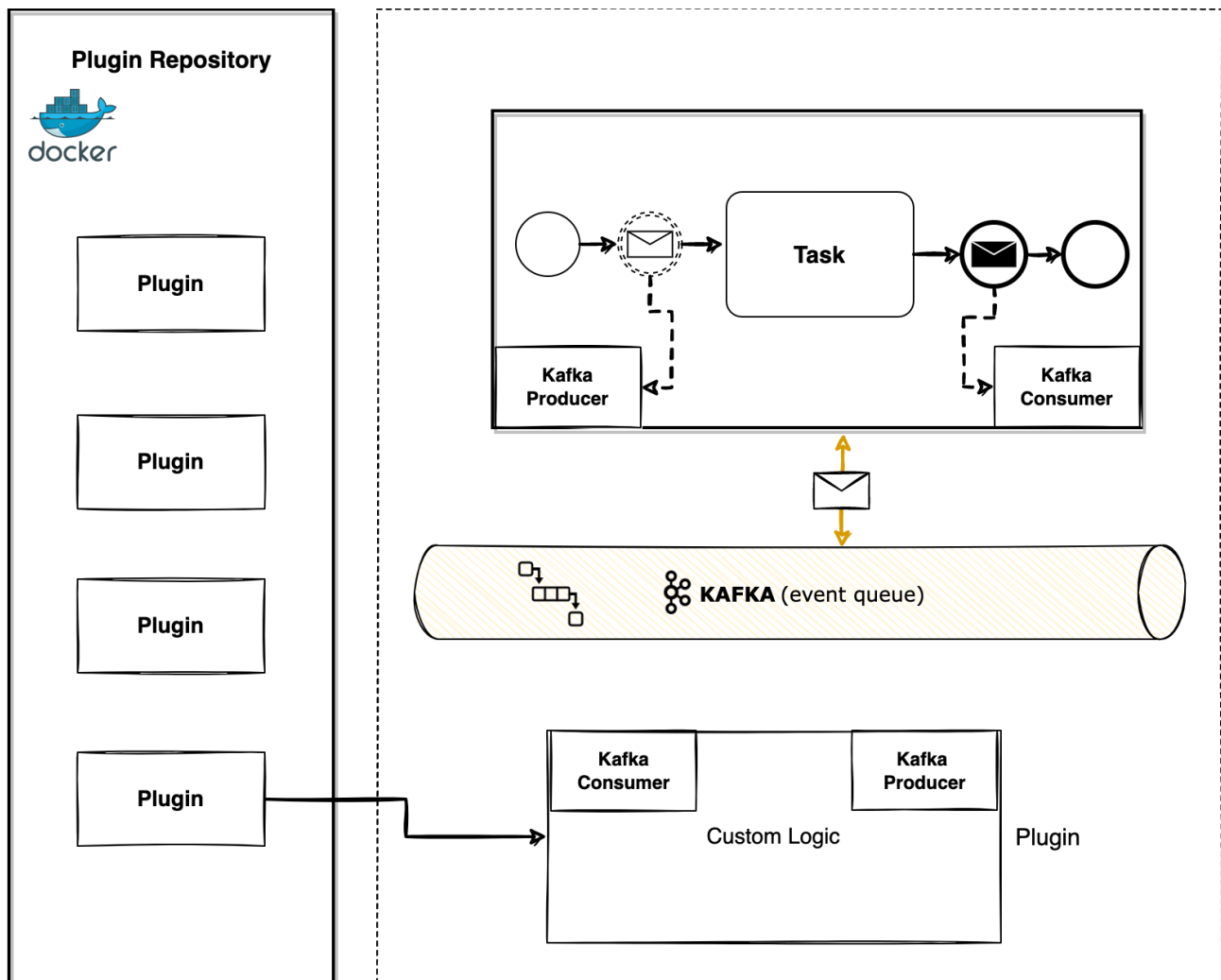
» Plugins

Integrations

Connecting your legacy systems or third-party apps to the FLOWX.AI Engine is easily done through custom integrations. These can be developed using your preferred tech stack, the only requirement is that they connect to Kafka. These could include legacy APIs, custom file exchange solutions, or RPA.

» Integrations

Was this page helpful?