Semester project

Implement a distributed system with a REST-based API, an ActiveMQ message queue and a JavaFX UI.

Scenario

You work in a company, that manages charging stations for electric cars. Every charging station keeps track of the customer that used the charger and the amount of kWh the customer used.

Your task is to build an application that generates an invoice PDF for a given customer.

Use JavaFX to create the UI for the application.

Use Java Spring Boot to create the REST-based API.

Use ActiveMQ to manage the message queue.

The workflow is as follows:

- You can input a customer id into the UI and click “Generate Invoice”
- A HTTP Request calls the REST-based API
- The application starts a new data gathering job
- When the data is gathered, it gets send to the PDF generator
- The PDF generator generates the invoice and saves it on the file system
- The UI checks every couple seconds if the invoice is available

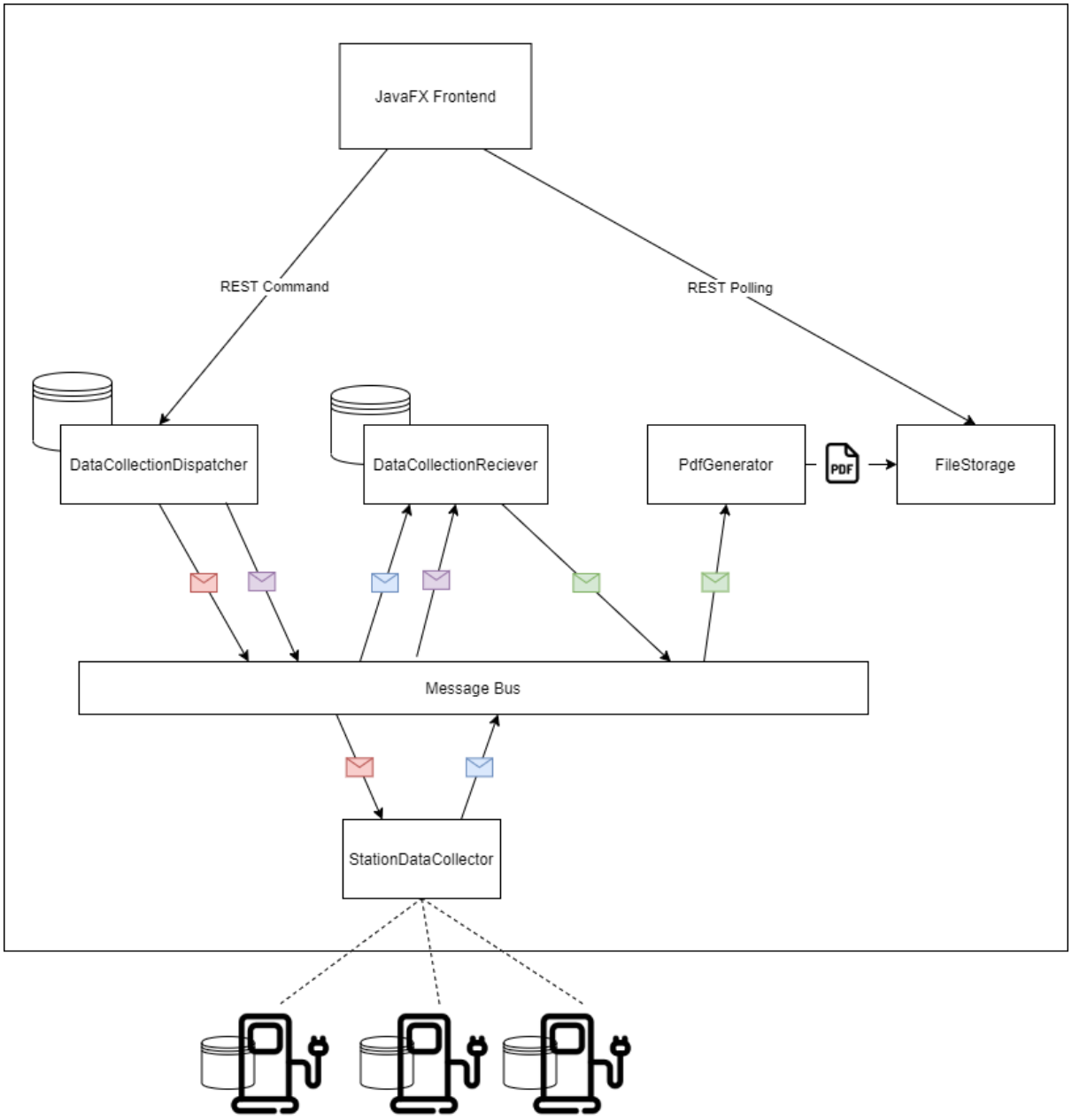
Specifications

There are four services that work on the message queue:

- DataCollectionDispatcher
    - Starts the data gathering job
    - Has knowledge about the available stations
    - Sends a message for every charging station to the StationDataCollector
    - Sends a message to the DataCollectionReciever, that a new job started
- StationDataCollector
    - Gathers data for a specific customer from a specific charging station
    - Sends data to the DataCollectionReciever
- DataCollectionReciever
    - Receives all collected data
    - Sort the data to the according gathering job
    - Sends data to the PdfGenerator when the data is complete
- PdfGenerator
    - Generates the invoice from data
    - Saves PDf to the file system

There are two API routes:

- /invoices/<customer_id> [POST]
    - Starts data gathering job
- /invoices/<customer_id> [GET]
    - Returns invoice PDF with download link and creation time
    - Returns 404 Not Found, if it's not available



Examples

Station Data (DataCollectionDispatcher)

id	station_id	available	latitude	longitude
1	1	true	48.205113	16.383420
2	2	true	48.238836	16.379323
3	3	true	48.196068	16.337239

Charging Station Data (StationDataCollector)

id	*station_id*	*customer_id*	*kwh*	*datetime*
1	1	5432	54012	2021-11-25 14:32:56
2	1	2344	46850	2021-11-28 08:14:12
3	1	5413	55445	2021-11-29 19:42:41