Intro

The objective of this test is to evaluate your:

- · fluentness in nodejs and express
- · data manipulation capacity
- · your sql understanding
- · capacity to create and deploy code

Mission

Create a nodejs express app that will import articles from a news website and save them inside a mysql database. And then the articles should be available in another endpoint. You are free to use whatever libraries you need, and add whatever features / controls you'd like. But bear in mind that more dependencies = more risks.

· This test is a showcase of your skills so do your best to display most of your skills.

· The estimated duration to complete the test is **1h** for the basic version and **2h** for a more complex version.

Tasks

· Provide an endpoint called `/api/articles/import?siteRssUrl`
· When this endpoint is called via method POST the node app will connect to the rss feed url provided in the `siteRssUrl` (we will use https://www.lemonde.fr/rss/une.xml OR https://www.theguardian.com/world/europe-news/rss for testing) -- will download the file, -- parse the xml -- will save all the articles in a database. The raw content will be saved in the imports table, and each article will be saved in the articles table (one article per insert)
· Provide an endpoint that displays the articles in the databases. `/api/articles`

Optional

· For each article in response of the endpoint `/api/articles`, send additionally the word with the most vowels (a,e,i,o,u,y) in the title. If two or more words have the same number of vowels, send the longest one.

Details

· The database table will be as follow :

Table `imports` that contains the history of all the time import was requested.

· id INT auto increment

· importDate DATETIME

· rawContent TEXT => the data imported (JSON)

Table articles (list all the articles):

· id int auto increment

· externalId: VARCHAR(500)

· importDate datetime

· title: TEXT

· description: TEXT

· publicationDate DATETIME

· description TEXT

· link TEXT

· mainPicture TEXT

Attention point, articles should not be imported twice. If the article already exists then do not create a new one, but update the existing one. You should be able to use the `guuid` as an `externalId` and identify duplicates with that.