

Indexed Text Search Microservice for Data Catalogs of The Mexican National Council for Science and Technology

RODRIGO CUELLAR | ALBERTO SANTIAGO MARTINEZ
BIBLIOTECA DANIEL COSÍO VILLEGAS | EL COLEGIO DE MÉXICO | MEXICO CITY

PROBLEM

The Daniel Cosío Villegas Library at El Colegio de México (BDCV) is developing a repository funded by, and as part the Mexican National Council for Science and Technology's (CONACYT) initiative to create a national aggregator.

CONACYT'S guidelines require that we utilize information from their national databases of authors, issues, rights, among others, via REST. The design of its API, presents severe limitations that put limitations in terms of search and retrieval.

Additionally, the BDCV team noticed that the CONACYT database contained various inconsistencies and lack uniformity between the names of authors of conacyt as well as the corresponding Internal and ORCID ID's.

Nevertheless, CONACYT requires that all grant recipients prove 100% interconnectivity and utilization of its national aggregator.

PROPOSED SOLUTION

Given these requirements the BDCV team designed a service that allows the following:

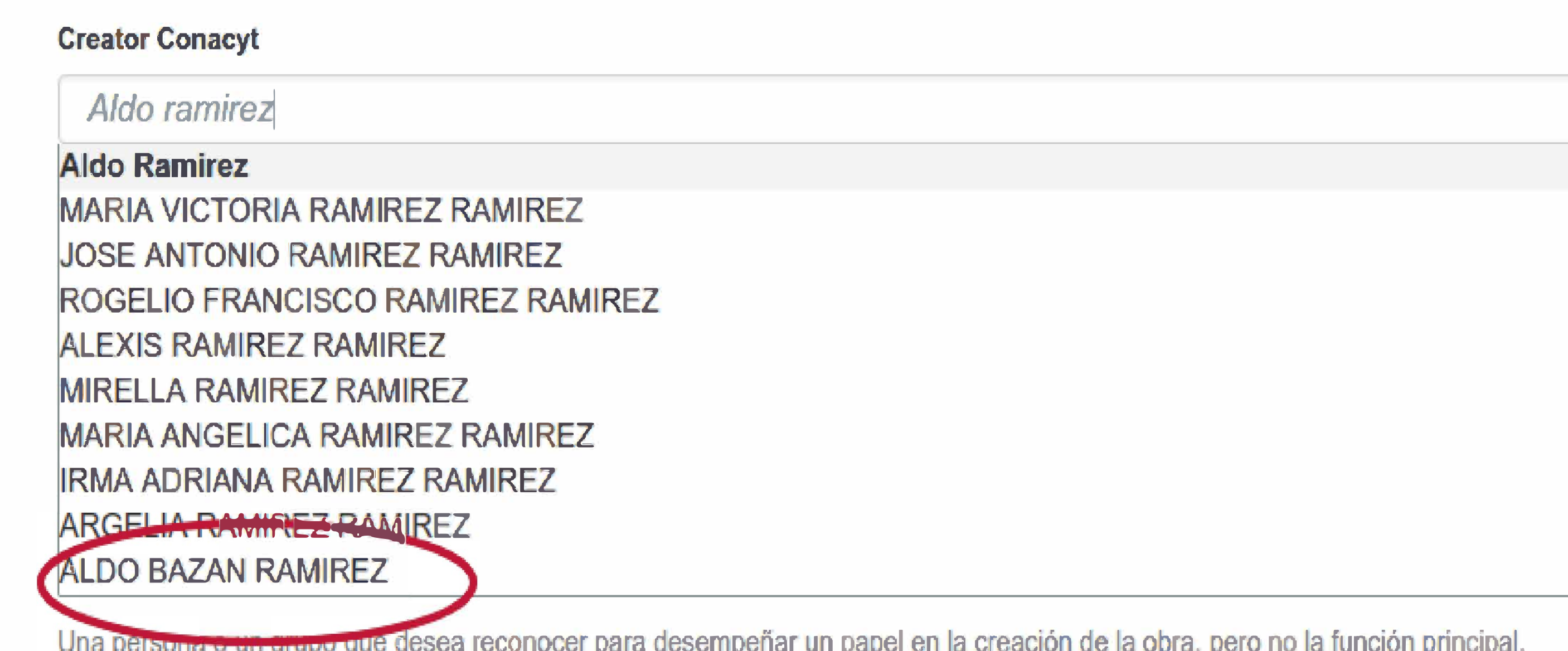
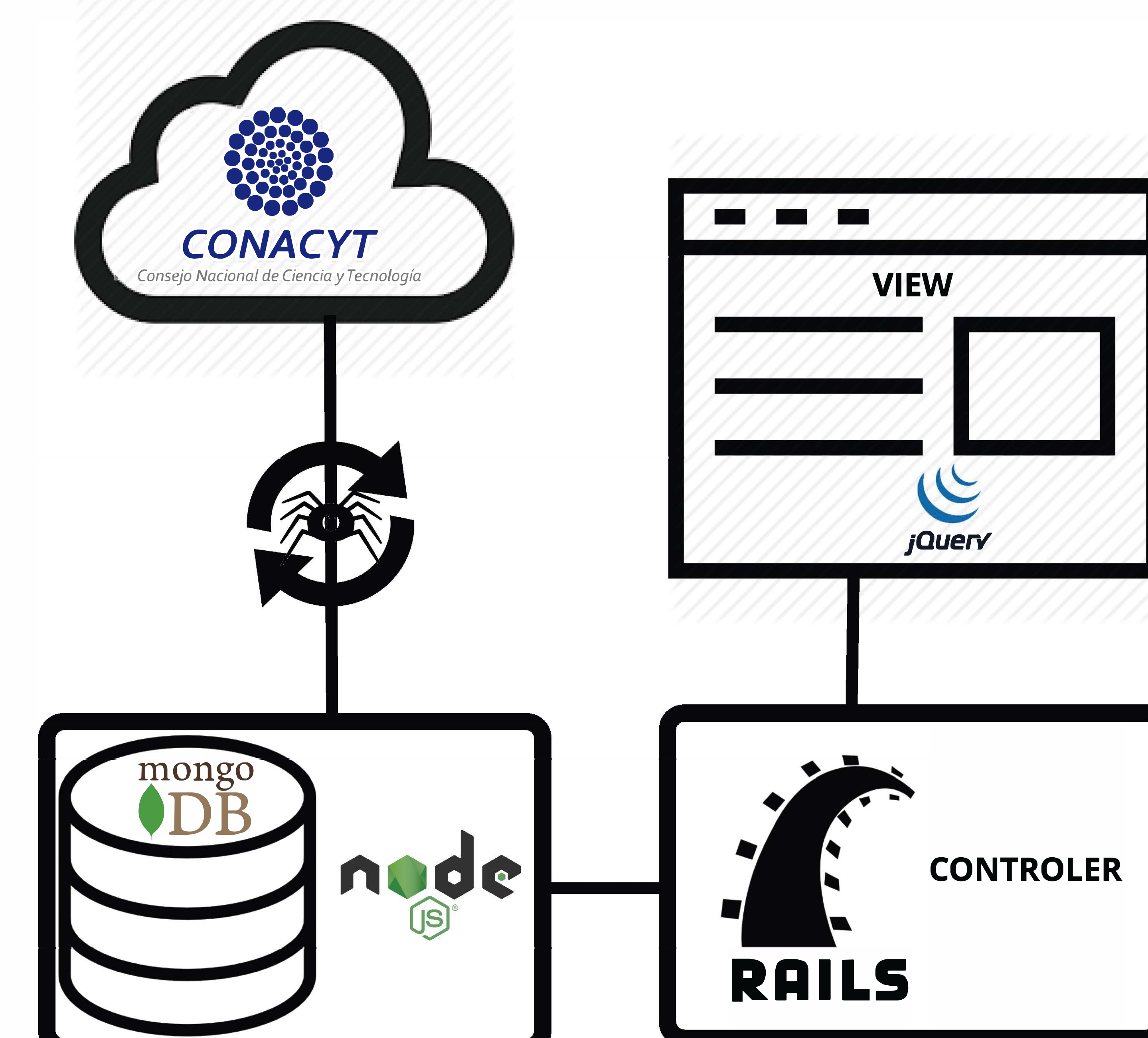
- Dynamic auto-fill search of Author names
- Upon selecting an author, the corresponding ID' fields (ORCID & CVU) would be filled in.
- Changes in their database would be automatically updated.

IMPLEMENTATION

For this purpose, a microservice was designed that would contain a complete copy of the CONACYT author database. MongoDB was used for its powerful text indexing capabilities, which allows searches to be carried out regardless of the capitalization or word order. Additionally, NodeJS was chosen to create the required EndPoints

WORKFLOW

- JQuery's easy-autocomplete plugin detects writing in the creator field in the client's browser.
- An Ajax request is triggered to a controller in Rails with the characters typed in the field.
- The Rails controller sends a Rest request to the microservice with the search parameters.
- The microservice returns the ten closest possible results to the search parameters.
- The Rails controller maps the response in a specific json for the autocomplete of the corresponding fields.
- The client's browser responds json and as soon as it detects the selection of the user, it reads the corresponding fields.



ACKNOWLEDGEMENTS

- Gabriela Montoya & the UCSD team
- Jennifer Lindner
- Samvera Community
- CONACYT