NYC Search Engine

Jobs API Endpoint

Problem

The City of New York has a jobs API they would like to add a front-end to. The API handles the ability to search for keywords or limit results you will be making requests to this API and displaying its results. On top of a front-end, the City would also like the ability to save and manage User information (without any need for authentication).

Billy

Age Range: 18-25

Occupation: Unemployed

Location: Brooklyn

“I really want to work for the city and wish there was a site to aggregate all available jobs for the city of New York.”

Jill

Age Range: 25-35

Occupation: NYC Public Health Inspector

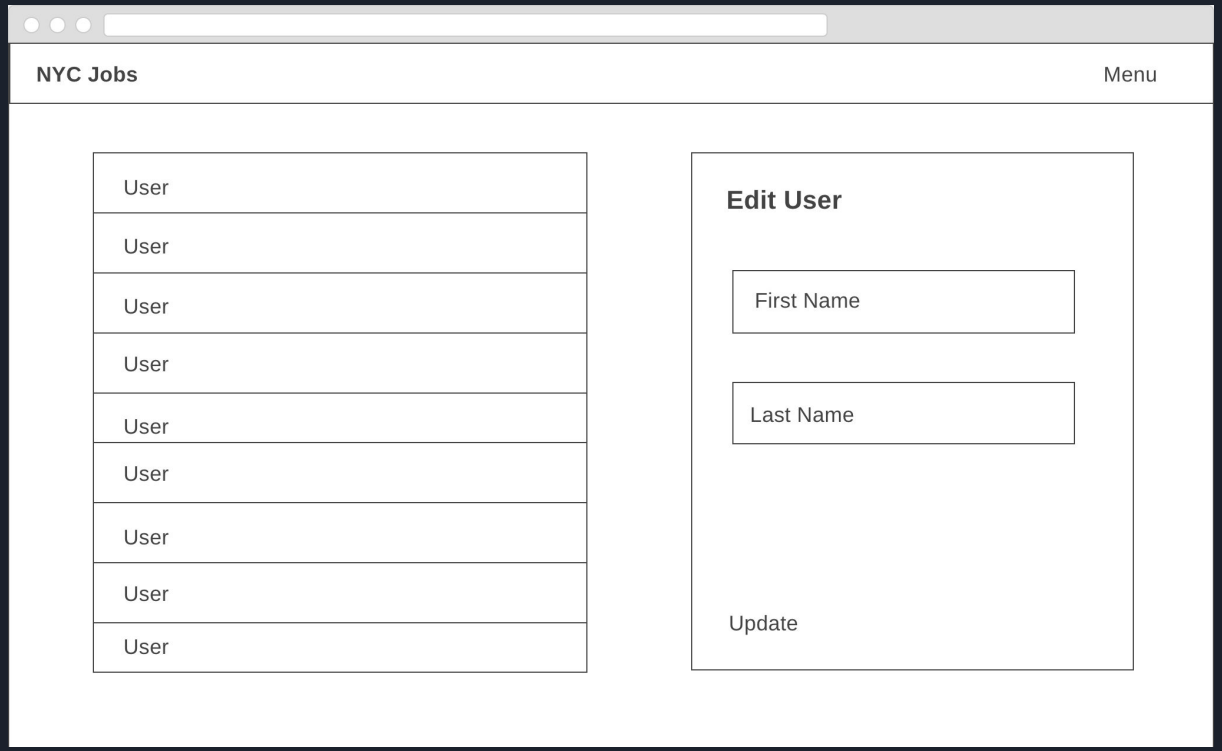
Location: Staten Island

“I would like to see any internal job postings in my department”

Problem (Cont'd)

The city of New York has a large amount of open jobs and there are many people available people to fill these spots. However, there isn't a site that can easily allow job searchers to find these jobs.

Wireframe (User Management)



Solutions

- Single Page Application using Angular
  - Responsive
- Spring controlling backend
  - Security
- Microservices
  - Isolation
  - Splitting services based on APIs (users, jobs, etc…)

Why microservices? (Instead of a monolith)

- Isolation
  - Failures
  - Development
  - Scaling

Microservices Example (Netflix)

Major Components of Netflix

Netflix Eureka – Service Discovery Server

Netflix Eureka allows micro services to register themselves at runtime as they appear in the system landscape.

Netflix Ribbon – Dynamic Routing & Load Balancer

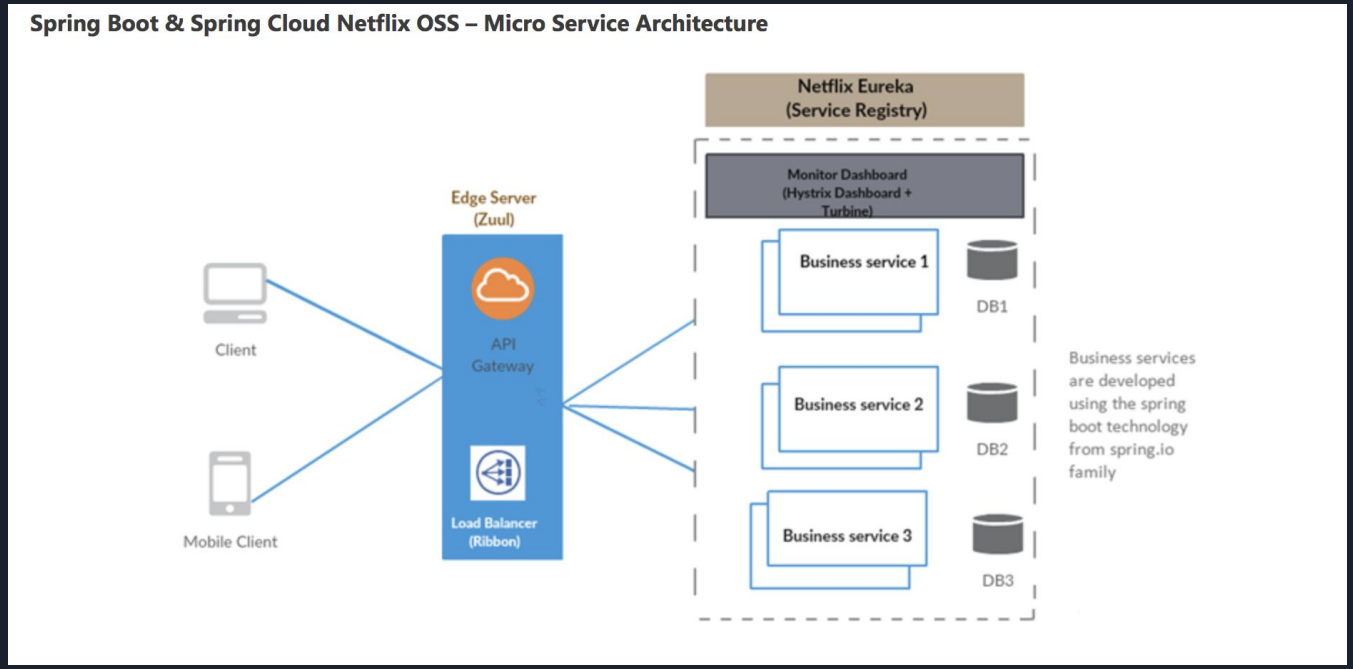
Netflix Ribbon can be used by service consumers to look up services at runtime. Ribbon uses the information available in Eureka to locate appropriate service instances. If more than one instance is found, Ribbon will apply load balancing to spread the requests over the available instances. Ribbon does not run as a separate service but instead as an embedded component in each service consumer.

Netflix Zuul – Edge Server

Zuul is (of course) our gatekeeper to the outside world, not allowing any unauthorized external requests pass through. Zulu also provides a well-known entry point to the micro services in the system landscape. Using dynamically allocated ports is convenient to avoid port conflicts and to minimize administration but it makes it of course harder for any given service consumer. Zuul uses Ribbon to look up available services and routes the external request to an appropriate service instance.

Source: https://www.optisolbusiness.com/insight/micro-services-architecture-spring-boot-and-netflix-infrastructure

Microservices Example (Netflix)



Source: https://www.optisolbusiness.com/insight/micro-services-architecture-spring-boot-and-netflix-infrastructure