CS 40700 Software Engineering Senior Project

Software Engineering students learn the Scrum Agile software engineering approach in CS 30700

CS 40700 is our capstone course

Students work in 4-6 person teams on their final project as college students

Important goal of the senior project is that they work well as a team and make professional presentations at each of their Sprint reviews

Teams typically produce excellent products using things they have been taught throughout their four years in CS and even learn how to use tools that they have not been taught in class

CS 40700 Software Engineering Senior Project

Examples:
AWS (Amazon Web Services cloud computing platforms)
DigitalOcean (cloud services to deploy and scale applications)
Bootstrap (open-source front-end Web framework)
Django (Python-based web framework)
Angular (TypeScript-based web application framework)
Node.js (open-source, cross-platform JavaScript run-time environment)
React (JavaScript library for building user interfaces)
Ajax (create asynchronous web applications)
Firebase (web application development platform)
PHP (general-purpose language designed for web development)
Symfony (PHP web application framework)
Cascading Style Sheets (language for describing the presentation of a document)
Restful APIs (provide interoperability between computer systems)
mySQL (open source relational database management system)
PostgreSQL (open source relational database management system)
MongoDB (NoSQL database program)
Electron (GitHub's framework for development of desktop GUI applications)
Google Cloud Speech-to-Text API
ONNX (Open Neural Network Exchange for interoperable AI models)

CS 40700 Software Engineering Senior Project

Team DZA, Fall 2018

Ben Maxfield, Brian Duffy, Adam Johnston,
Terry Lam, Christian Lock, Jalen Smith

Project Coordinator Alina Nesen

Faculty Project Owner Prof. Petros Drineas

DZA

API Data Version History

A software engineering senior project

What is an API?

Application

Programming

Interface

A set of functions or procedures used to access and communicate with other applications and services.

We use **HTTP requests** as our main method of communication when requesting resources from different APIs.

What is an HTTP Request?

Hypertext

Transfer

Protocol

HTTP is a protocol that defines how message or request for a resource is defined.

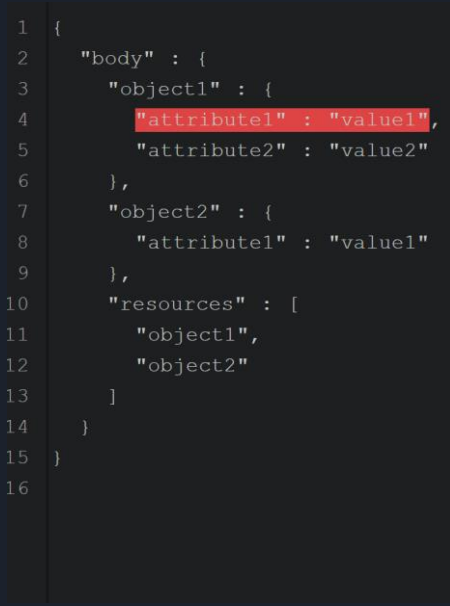
Contain a method, a URL, and parameters

GET https://www.googleapis.com/gmail/v1/users/me/messages?q="in:sent after:2014/01/01 before:2014/01/30“

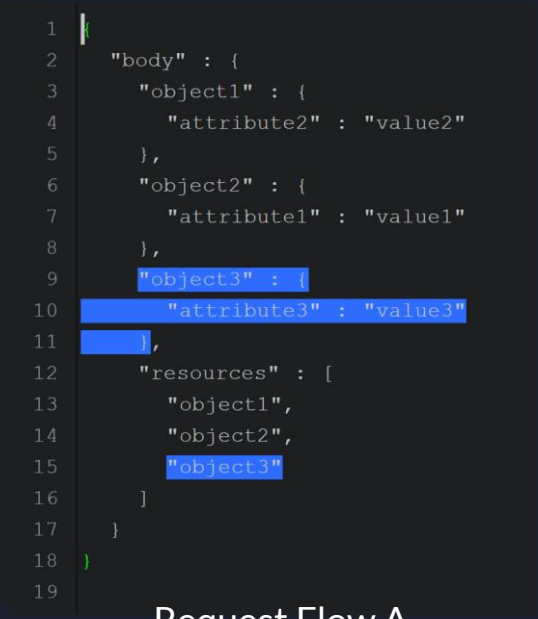
*An HTTP request to Google’s Gmail API, which is requesting all messages sent between two dates

How DZA works

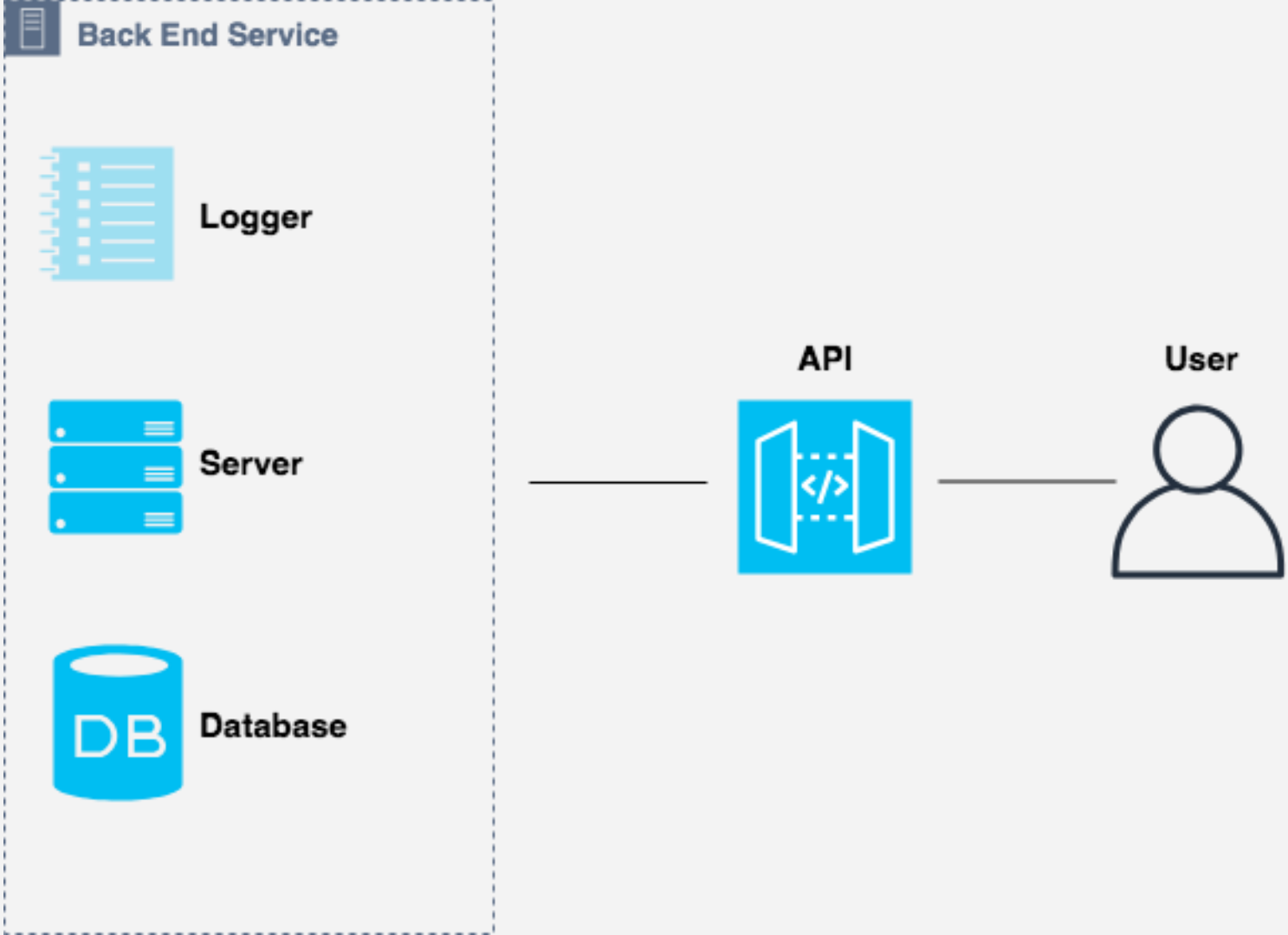
1. Users create a “flow” of API requests, and feed the responses of one request into the next.
2. These flows are executed on our server and the resulting response is saved.
3. Responses can be compared to previously executed flows to look for differences.
4. Depending on the user’s needs, flows can be executed on a regular schedule

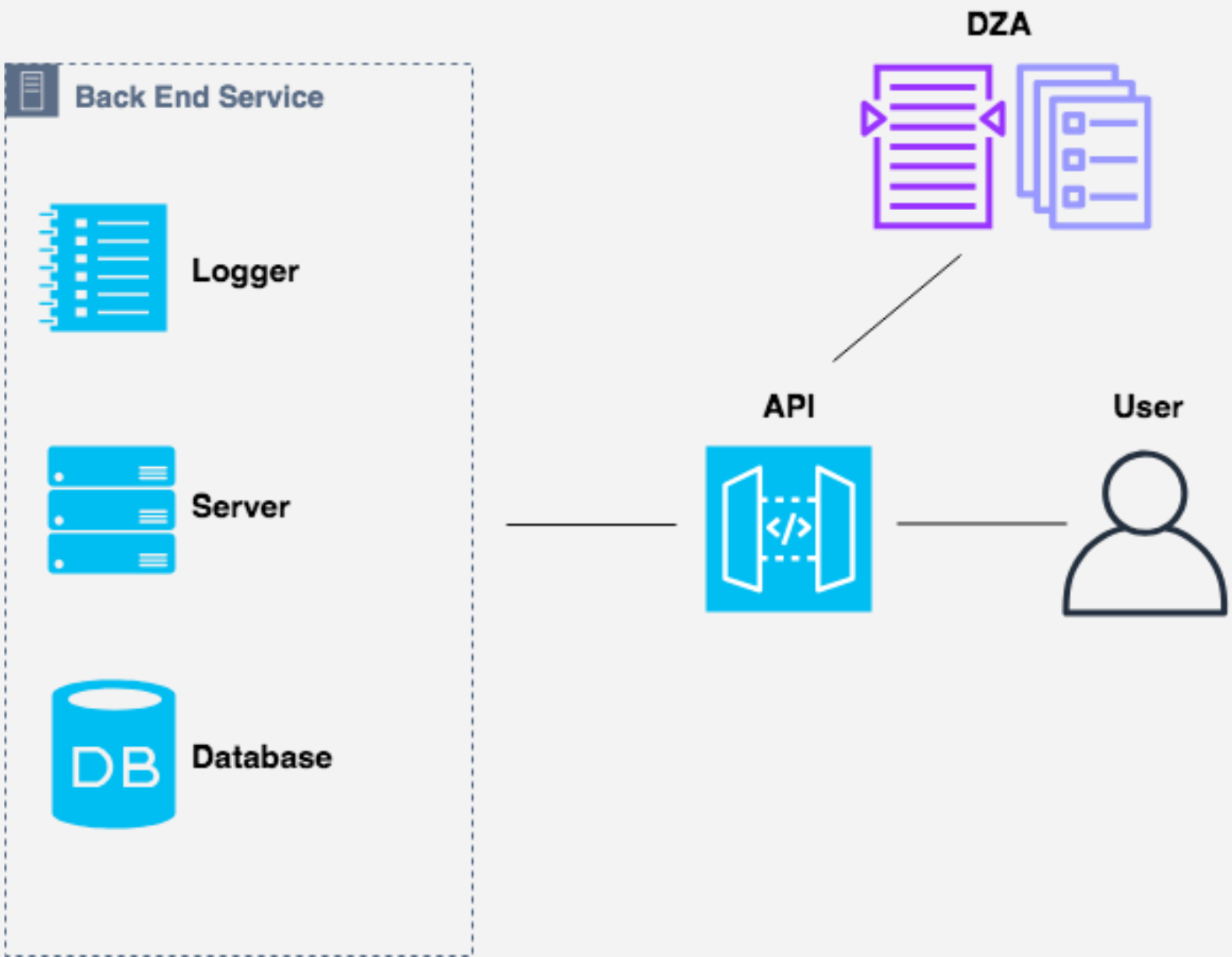


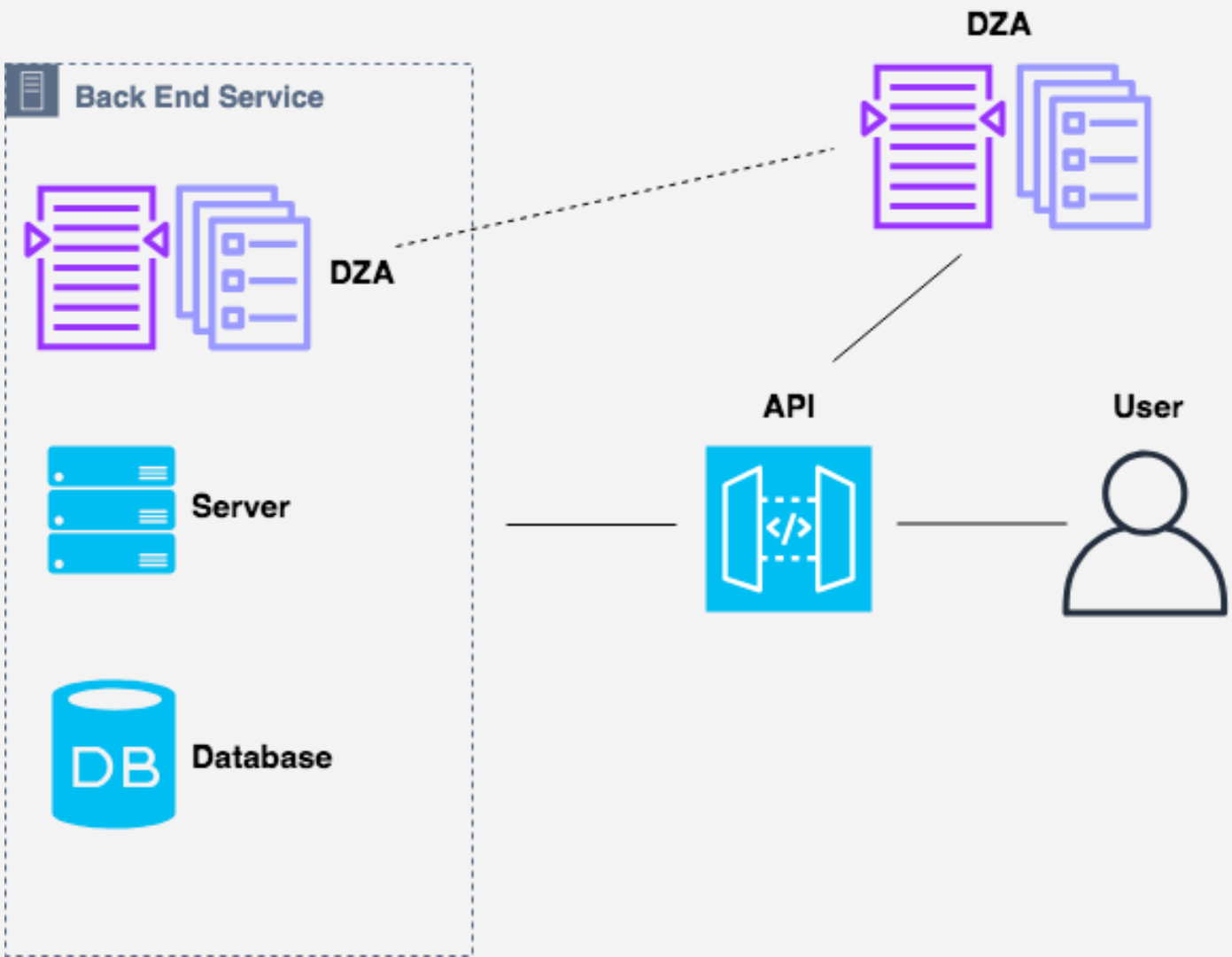
Request Flow A
2018-12-06
22:00:00



Request Flow A
2018-12-07
22:00:00


Back End Service
Logger
Server
DB
Database
API
User


DZA
Back End Service
Logger
Server
DB
Database
API
User


DZA
Back End Service
DZA
API
User
Server
DB
Database

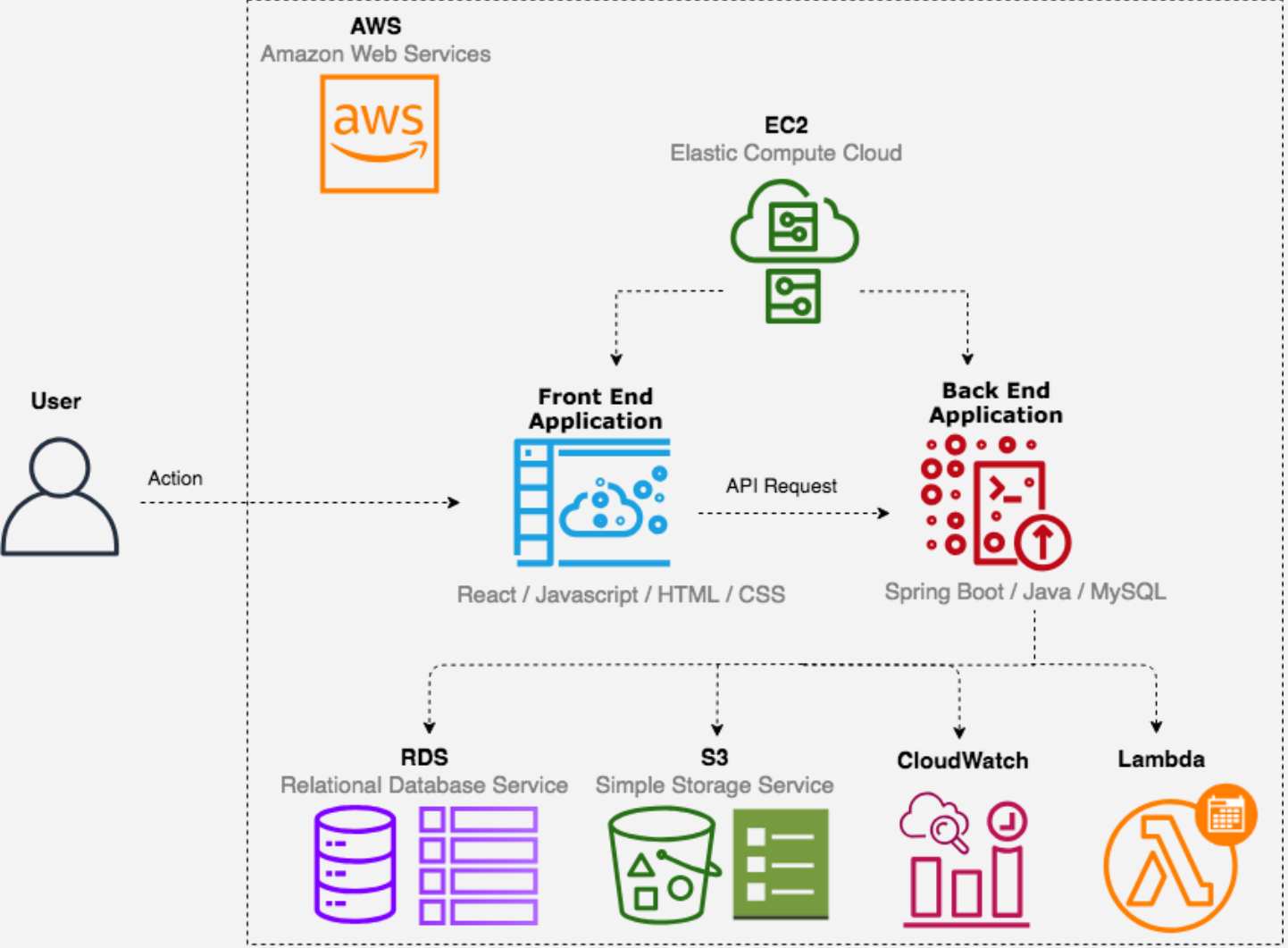
DZA is *Versatile*

DZA fills a service gap for developers

- Scripting - makes it easy to automate actions
- Operations - tracks changes to production data
- Testing - logs responses to test requests

DEMO

Application Architecture Overview



Q&A