Final Project: Project MIX
The final project for CS 240 this semester is a course-wide microservice called "Project MIX".

- **Frontend**
  User provides GPS coordinates as input, the client's web browser makes a **POST /gps** request with the GPS coordinates, and the response to that POST request is data about the GPS coordinates.
  - A very basic frontend is provided.
            ...this can/should be extended later!

- **Middleware**
  Receives the **POST /gps** request from the frontend, retrieves information about the GPS coordinates from bandend services, and returns all of the data back to the frontend via a formatted response.

  Right Now: Every group is designing a middleware design and implementation right now. One design will be chosen to be used at the official class-wide MIX design based on your designs!

- **Bandend**
  The functional design of MIX requires that all data about the GPS coordinates is fetched by microservices called "Information Microservices" (IMs).

  Many possible examples:
  The city it's located within (ex: Urbana, IL); the state it's located within (ex: Illinois); the weather forecast; the most popular restaurant within 1 mile; the walkability of the city/area; the most played song within the city/area; the state bird; the sales tax rate; the nearest Starbucks coffee shop (or any coffee shop); the zip code; and any number of millions of other things.

  Right Now: Every group is designing a collection of IMs that will work with the class-wde microservice.

Cloud-Scale Content Distribution
As we deploy to a cloud-scale, there are three different types of content we need to deliver:
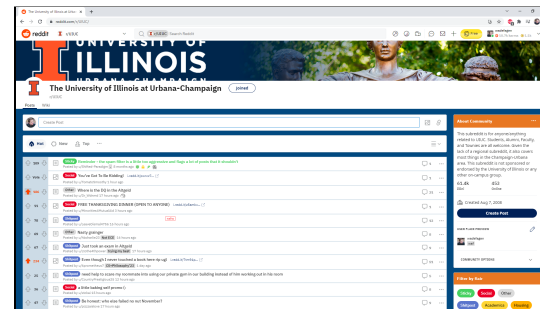
- Static Content:

  Usually cached for...?

- Universally Rendered Content:

  Usually cached for...?

- Dynamic Content:

  Usually cached for...?

Example: https://reddit.com/r/uiuc
reddit.com is a community organized into subreddits that focus on particular topics. r/uiuc/ is the subreddit for the UIUC community:

https://reddit.com/r/uiuc

Static Content?

Universal Rendered?

Dynamic Content?

Motivation: Does the user need to visit **our server** for content that is served to all users?

- If not, what service could we rent?

- What advantages can we get by renting caches?

Content Delivery Networks (CDNs)
A Content Delivery Network (CDN) is a system of many servers physically located in geographically diverse locations.

- All CDNs must have a source for the content. This source is known as the _____.

- All CDNs have many caches called _____.

- Once the data from the origin is in the CDN cache, the CDN can serve this content. This provides five key benefits:

  1.

  2.

  3.

  4.

  5.

Geographical Locations of Edge Servers

CloudFlare CDN Locations	AWS CDN Locations
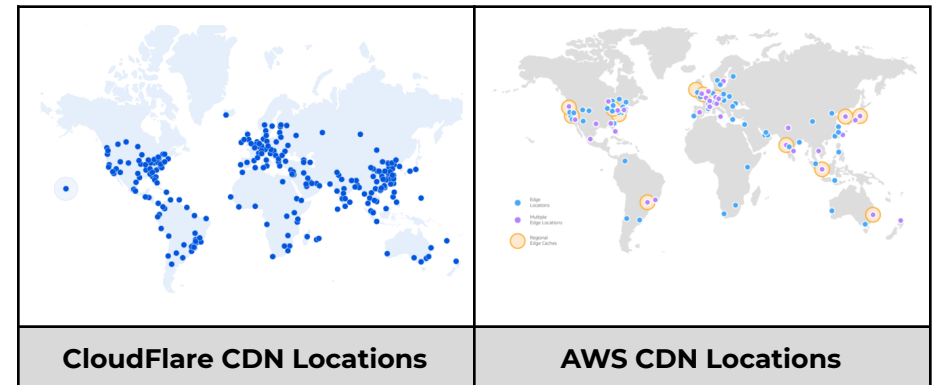
How do we use CDNs?
Recall that looking up the IP address for **cs.illinois.edu** requires a series of DNS requests:

1. Ask a root name server for the NS records for "edu."
   ⇒ *IP address for the TLD name servers for "edu." returned.*

2. Ask a "edu" TLD name server for the NS records for "illinois.edu."
   ⇒ *IP address for the SLD name servers for "illinois.edu." returned.*

3. Ask a "illinois.edu" SLD name server for the NS records for "cs.illinois.edu."
   ⇒ *IP address for the sub-domain name servers for "cs.illinois.edu." returned.*

4. Ask a "cs.illinois.edu" sub-domain name server for the A (or AAAA) records for "cs.illinois.edu."
   ⇒ IP address for web server for cs.illinois.edu returned.

Q: Could we program a specific DNS server that was geographically aware to return different data?

...what should the TTL of these A records be?