Week 3 Day 1

REST Webservice

Webservices: REST

REVATURE

Representational State Transfer

- Style/architecture that outlines communication
- Must follow these constraints
  - Uniform interface
  - Client-Server
  - Stateless
  - Cacheable
  - Layered
  - Code on demand

REST: Resources and Naming

REVATURE

- Resource refers to any information
  - Identify them with URL's
- REST service naming conventions
  - Nouns to name resources
  - Use plural for collections of resources
  - Use path parameters to specify a document
  - Use path structure to create hierarchy
  - Identify stores of resources managed by the client
  - Use query parameters for filtering
  - BE CONSISTENT!

Webservices: Authorization vs Authentication

REVATURE

Authorization

- Determine users access level

Authentication

- Determine/verify user is who they say they are

Source: https://www.ssl2buy.com/wiki/wp-content/uploads/2019/04/authentication-vs-authorization.jpg

Session Management: Cookies

REVATURE

Key value pairs of information shared between client/server

- Sent in the header
- Verify they are present

Non-persistent

- Expires when the browser is closed

Persistent

- Expires when you decide (ex. when a user logs out)

Session Management: Cookies

Set a cookie with document.cookie = ‘the cookie’

- Only store none important information
  - Cookies can be seen by everyone
- Cookies only work if they files are being hosted by a server
  - Static locally served files will not allow cookies

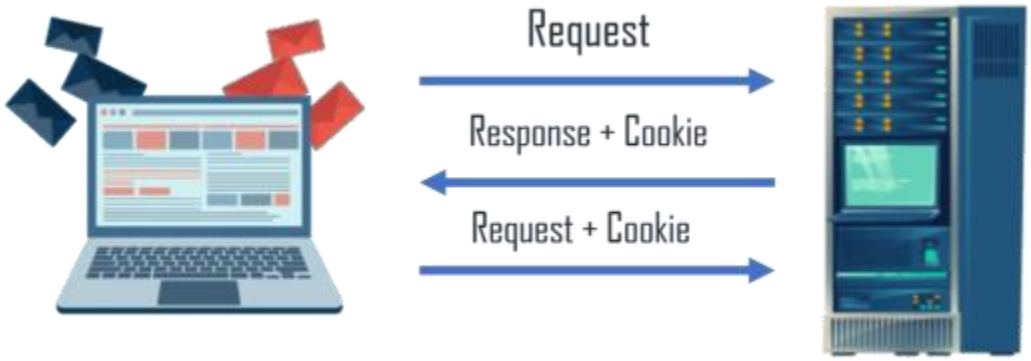

Image source: https://www.edureka.co/blog/wp-content/uploads/2019/08/cookie-528x206.png

Session Management: Sessions API

REVATURE

An API which allows you to identify and store user information based on http requests

- Built in session management with Javalin
- Access these through the context
  - context.req.sessionAttribute(“key”,”value”)
  - context.req.getSession().sessionAttribute(“key”)
  - context.req.getSession().invalidate()