

WEB APPLICATIONS

Crash course on web stuff

SERVER

(BACK-END)

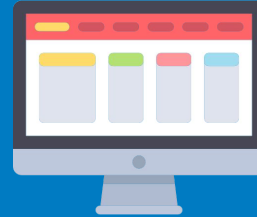
SOMEONE ELSE'S MACHINE



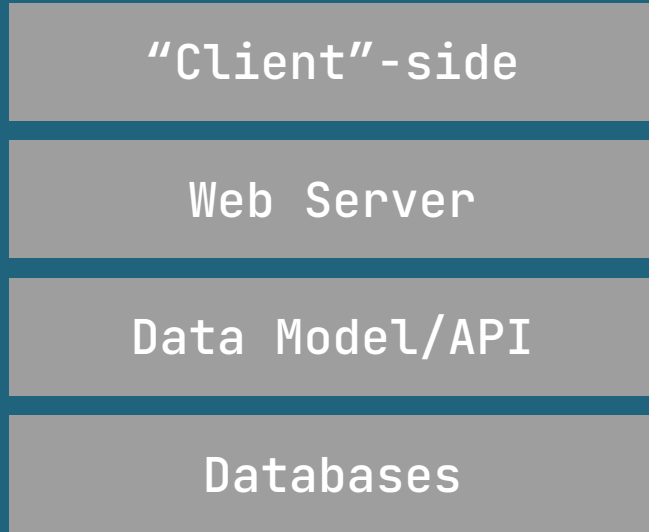
CLIENT

(FRONT-END)

YOUR PC



The “Stack”



The "Stack"

"Front End"

Presentation layer

Data Model/API

"Back-End"

Data



Flask
web development,
one drop at a time

{JSON}



Crash course on web stuff

SERVER

(BACK-END)

NOT ALWAYS SOMEONE ELSE



BUT

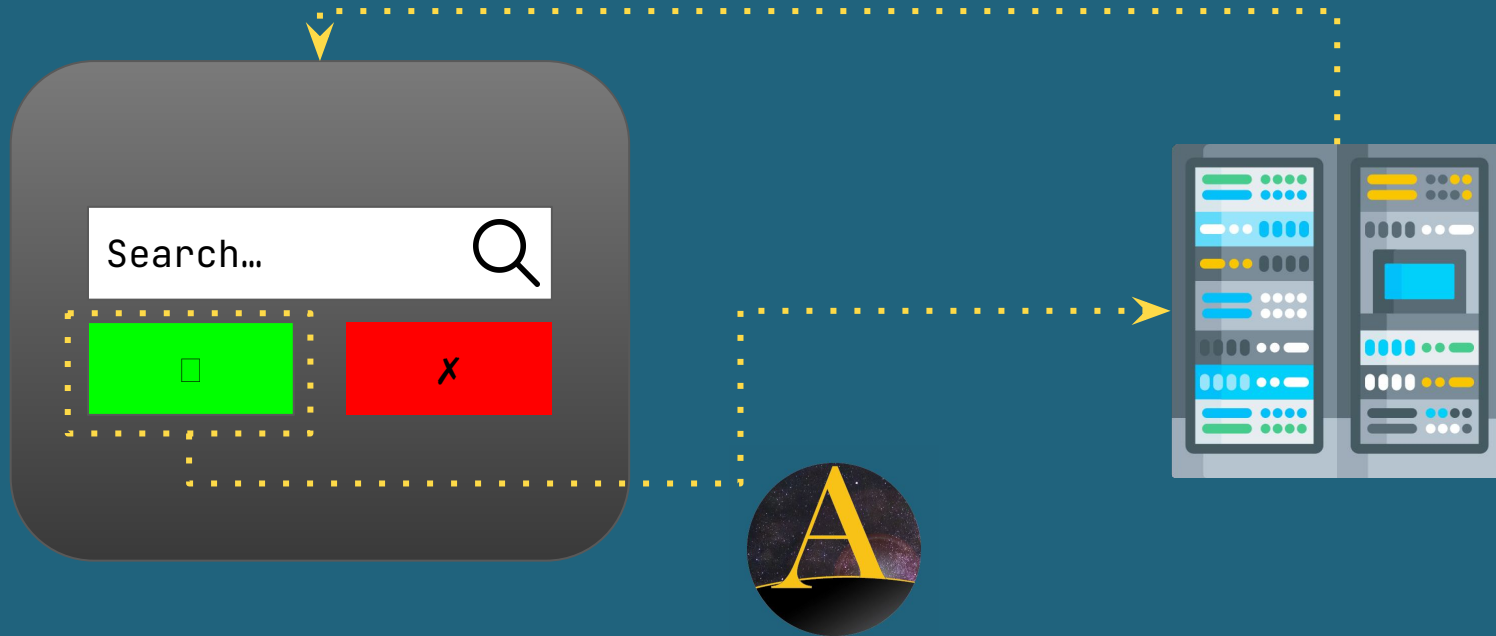
**BACK-END CAN
ALSO BE MY MACHINE?**

imgflip.com



APIs

Mechanism to answer web requests and provide data to a given application requesting it.



APIs

Are constructed of “endpoints” (i.e., URLs) which respond to given web “verbs.”

COMMON WEB “VERBS”

GET

POST

PUT

DELETE

PATCH



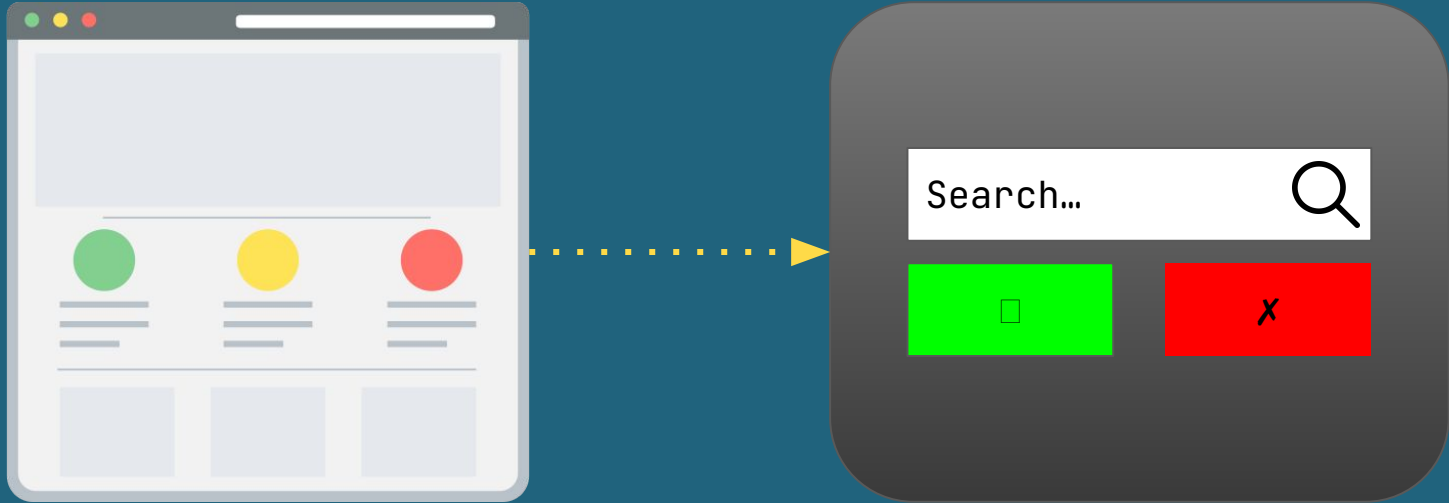
APIs

COMMON WEB “VERBS”

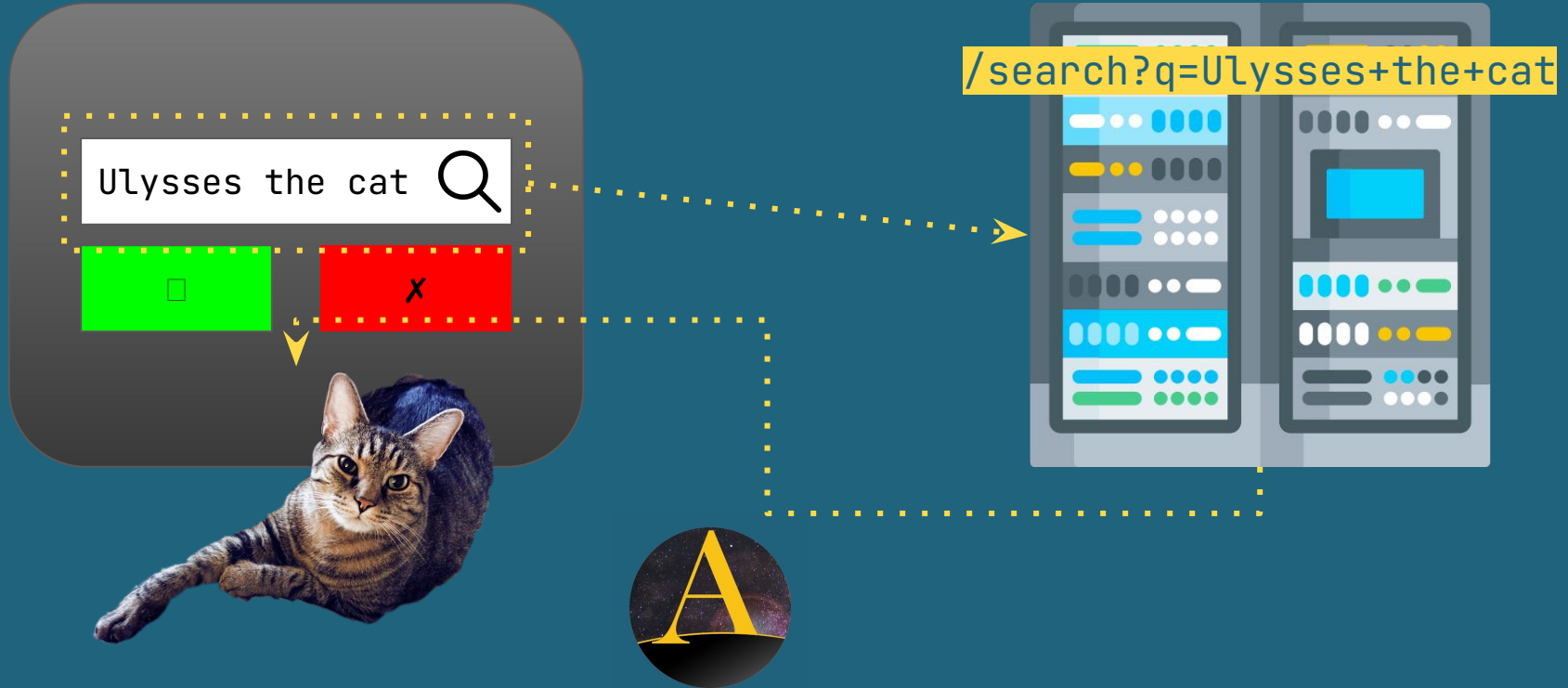
GET	Fetch a representation of a thing
POST	Submit data motivating an application state change
PUT	Replace representation with new data
PATCH	Replace part of representation with new data
DELETE	Delete representation of a given resource



GET Request

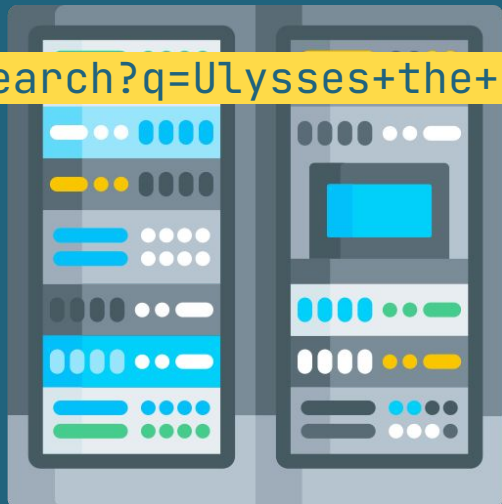


POST Request



POST Request

/search?q=Ulysses+the+cat



Param: Ulysses+the+cat



Route: /search



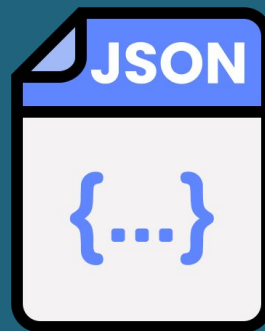
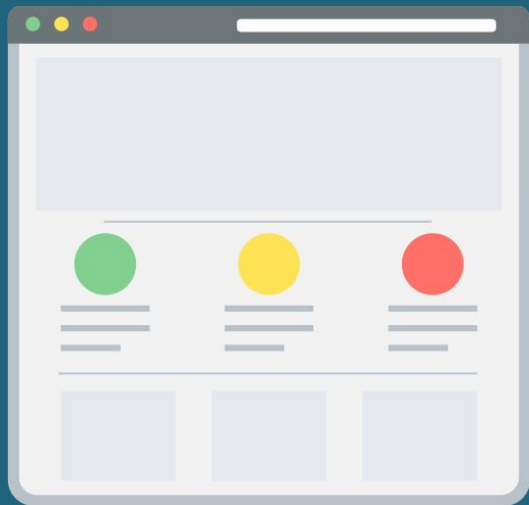
POST Request



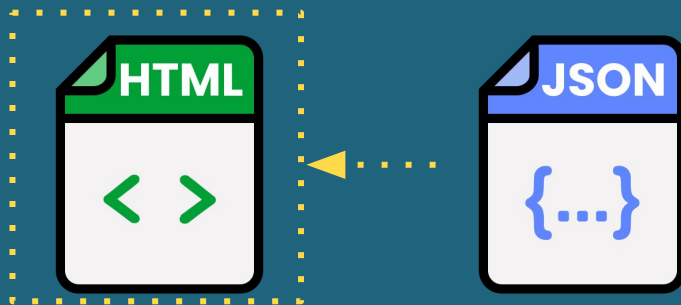
`/search?q=Ulysses+the+cat`



POST Request



REST API



Representational State Transfer

