Introduction to BE Development + Flask

Randy Truong

rtruong@u.northwestern.edu

February 2, 2024

Outline

1. conceptual intro to be dev
   - look into full stack dev
   - be dev narrowing
   - be dev narrowing
   - be dev narrowing
   - be frameworks
   - flask demo
   - flask demo
   - flask demo

Introduction to BE Development

full stack application explanation

introduction

at its core:

- **frontend.** a html, css, js program that (mostly) just renders things to the screen.
- **backend.** a *separate* program that allows us to offload these computations to another machine

full stack application explanation

introduction

at its core:

- **frontend.** a html, css, js program that (mostly) just renders things to the screen.
  - buttons on the website will allow u to perform different computations + functionalities
- **backend.** a *separate* program that allows us to offload these computations to another machine
  - can be done in most languages (primary ones are python, js, and java)
  - primary function is to be an *event handler*
  - frontend will communicate to the backend, to which the backend will respond

what is be dev?

backend dev is a request handler

- the client tells the be application to do something $\rightarrow$ be application does the thing $\rightarrow$ be application reports back to the client
- bank metaphor (but before computers idk)

what is be dev?

backend dev is a request handler

- you, a customer of the bank, wants to retrieve your account balance
- you talk to a teller and indicate to them that you want to retrieve your balance
- teller tells internal staff about ur request and they go into the bank vault and count ur money
- internal staff tells the teller ur account balance
- teller tells u ur account balance

what is be dev?

ok cool, but how does it work?

- **remark.** backends are *separate applications* from the frontend
- they are apps that are designed to *listen* for requests
  1. think of the backend as basically just being a dictionary for mapping requests to functions
     1. receive request from client ("i want x")
     2. checks for an entry for this type of request
     3. respond to the client in either outcome

what is be dev?

thus, the "format" of a backend application is as follows:

- setup listening (usually done by the package/framework)
- define your "request dictionary". for each request:
  - define what a client should "say" in order to demonstrate that it wants to do something (key)
  - define a function that executes following that request (value)
    - we can either perform that computation within the be application itself
    - or just send ANOTHER request to another application

what is be dev?

thus, the "format" of a backend application is as follows:

- setup listening (usually done by the package/framework)
- define your "request dictionary". for each request:
  - define what a client should "say" in order to demonstrate that it wants to do something (key)
  - define a function that executes following that request (value)
    - we can either perform that computation within the be application itself
    - or just send ANOTHER request to another application

frameworks?

in order to do the things i just described is *complicated*. we can do them in c and python without installing anything, but it sucks and there's a lot of places to fail

- frameworks abstract away all of the complex code for sending requests
- using frameworks, we can just think of a be application as just being a dictionary

flask?

flask is a lightweight be framework in python

- super easy to use
- doesn't any config (think react config)
- super intuitive

flask?

flask is a lightweight be framework in python

- super easy to use
- doesn't any config (think react config)
- super intuitive

flask?

again, a flask application is mostly just going to be a response handler

- **routes** create a mapping between a request type and a response
- how to test it?
  - we can either send a request from browser (where browser is the client)
  - or make a python program that sends requests (where this separate py program is a client)

flask demo time