

# Web, Python, Flask, And You

Tucson Python + ResBaz AZ 2023

Poppy Argus 2023-04-18

# Outline?

- “Messaging App” stream of consciousness? Pt1, pt2 ... ptN
  - Whole talk is “diff 1 with 2” then “2 with 3” etc.
- Flask, docs, web
- Opinions, could be gettin’ it wrnog in front of an audience
  - Please let me know! I’m happy to be wrong. (After, please! – take a note, bring up then)
- Expository
  - Not a class
  - Details left as exercise for the reader
- You will know how to at least find out more



# Start At The End

- Demo here
- Repo: <https://github.com/poppyargus/> and click on py-talk

# pt1

- Add app
- I want to build an app, Hello World
- <https://flask.palletsprojects.com/>



# pt2

- Need app to: add “messaging”, “user”
- I want to build a messaging app; what parts?
- Flask quickstart, route w vars
- App code

# Intermezzo – Makefile, venv, pip...

- `make clean setup lint`
- Did you think I remember all these commands?
- `cat Makefile`



# pt3

- Need msg + user to: add “HTML”
  - “find and replace”
  - Explicitly not a template yet
  - We’ll talk about jinja in a minute, GOSH
- Obvs not secure, but whee

# pt4

- Need “HTML” to: add login & Session
  - Session: a LOT of security
  - Learn how this works, seriously, or expect pain
  - <https://flask.palletsprojects.com/en/2.2.x/quickstart/#sessions>
- More secure...



# Intermezzo - templates

- <https://flask.palletsprojects.com/en/2.2.x/tutorial/templates/>
- Boo Jinja: Just look at it. Same HTML. Yikes.
- Boo f-string: No conditional, loop, struct.
- Want
  - Push vs pull, want data vs code
  - Some conditionals, data-based
  - Multi-language. No py in HTML!
- What could we use instead?

# pt5

- Need HTML + Session to add: Mustache Templates
  - <https://mustache.github.io/>
  - Via chevron, because it's what mustache links as the py impl
  - <https://github.com/noahmorrison/chevron>
- Can also run on command line
- data + template
- Based on Google's ctemplate



# pt6

- Need mustache complexity to necessitate: add testing
- Pytest
- How extensive?
  - Much like the line about flossing...
  - Here: crux? `get_body_template`
  - Avoid testing the language, libraries

# pt7

- Need testing to: add storage abstraction
- Can't get this wrong
- Starting to see how each small chunk testable
- Wild parts should be just glue code
  - Still end-to-end test, automate early, not too early



# Intermezzo – Data Mapper

- One example of a data mapper architecture
- Alternative to ORM (Object Relational Mapper)
- Great stories about this in my career, great source of reliability, dev speed, empowerment
- Multi-impl – test, dev in-mem, dev on-disk, prod sql, prod noSQL
- Not without drawbacks (ie, transactions across multiple large complex joins)
- Fowler: <https://martinfowler.com/eaCatalog/dataMapper.html>

# pt8

- Not implemented; idea:
- Need storage abstraction to: add persistence
- Can now swap – disk at run, mem for dev
- StorageSqlite
- `pytest.parametrize` to run same tests with multiple implementations



# pt9

- Not implemented; idea:
- Need persistence to: add message sending
- User page: form for user / message
- Gitignore run dir

# Have to Stop Now

- (ad infinitum)
- (draw the rest of the \_\_\_\_ owl)
- (etc.)
- (how long did you think I was gonna talk)



# This talk

- Talk about flask
- Talk not at all about flask (so what is it about?)
- Might sound like criticism, but only web framework I have used professionally and not found worse than nothing.
- Don't have to “think in Flask” yay!
- Don't have to follow the manual.

# Now, you...?

- Know flask?
- Know where to find more?
- Know why you want to use it?
- Know mustache & why maybe not jinja?
- No? Ask me after!
- `poppyargus2023+resbaz@calmspark.org`