

Refactoring an R Shiny Application in RStudio

Nadia Antony

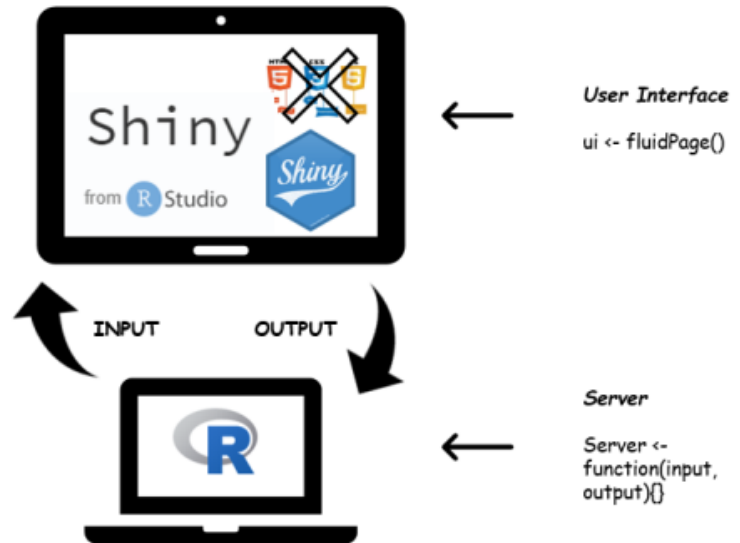
Analytics @ OAITI | nadiantony@hotmail.com

An R Shiny Application

Web Page powered by an R session

shinyapps.io - host your apps

[Shiny Gallery](#)



Inspiration: Refactoring an RShiny app

- Data Analysis is highly interactive but also collaborative
- Tools like RShiny are great for prototyping but can quickly become complex

Helps to follow a process

- Small changes to structure code effectively
- Done before adding new features or after a release

Inputs mapped to
Outputs
Flow between modules

Running the App

We should ideally be able to run the code directly from [GitHub forked repo](#)

Tree: 400b74228f ▾

New pull request

Create new file

Upload files

Find file

Clone or download ▾

lotusxai update remove dummy

Latest commit 400b742 on Feb 25

SQL_scripts

update for demo

last month

automation

update for demo

last month

pet-records-app

update for demo

last month

README.md

update for demo

last month

SQLite_conversion.R

update for demo

last month

final_data_model.jpeg

update for demo

last month

get_data.R

update for demo

last month

So does this work?

```
shiny::runGitHub("NadiaAntony/pet-days", subdir = "pet-records-app",
```

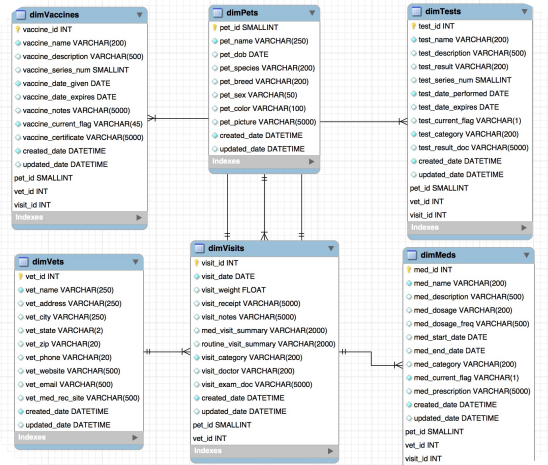
```
Error in library(fs) : there is no package called 'fs'
```

install the required package on my local using `install.packages()` and then try another run:

```
Error : no such table: dimPets
```

Primary Tasks

- mental model of architecture
- dummy data using data model provided



- setup script to propagate the data to SQLite DB `setup_db.R` [commit#](#)
- setup script to provide a `requirements.txt`, and install packages if missing `run_requirements.R` [commit#](#)

Output

Pet Records

Select pet:

- ☐ Mylov
- ☐ Angel
- ☐ Tika

 Weight History (lbs.)

Built with  by  Studio

 Medical History

 Vaccine History

 Medication History

 Vet History

Medical History and Tests Timeline

Click an item to view more details or test results (where available). The information is shown below the timeline.

☐ Show routine visits



SHOW ALL ITEMS

RESET TO DEFAULT

Reinforcing Primary Tasks

Reinforce these changes through

- setup instructions and documentation in the README

setup

```
git clone https://github.com/lotusxai/pet-days.git
cd pet-days/

Rscript run_requirements.R
```

run_requirements.R installs the necessary packages, runs setup_db.R and tests the setup. Note: The setup_db.R script contains the view and table creation scripts. You can check the output in log files in the assets/log/ folder.

running the app

If all the tests pass, the following command launches the shiny app in your browser.

```
R -e "shiny::runApp(launch.browser = TRUE)"
```

- unit tests
- logging
- continuous integration

Unit Testing

Test Expected Outputs:

Function outputs are checked using `testthat` package

test scripts created for:

- installing and loading all libraries
- data propagation, reading and loading [commit#](#)

tests called after `setup_db.R` in `run_requirements.R` [commit#](#)

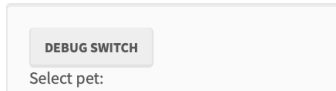
Log file

Simple logging of the `run_requirements` file if done using the `sink()` command to redirect outputs.

```
logfile <- file(file.path(logs_dir, "run_requirements.log"), open = 'a')
sink(logfile, append = TRUE, split = TRUE, type = "output")
```


Next Steps

- Using browser() to debug issues like Invisible UI Outputs
Pet Records



- Shiny best practices and modularizing code
- Adding in enhancements, e.g, data writes from the app

Thank You!

Find me on

- GitHub: github.com/NadiaAntony
- LinkedIn: linkedin.com/in/nadiaantony
- Medium/Twitter: [@NadiaAntony](#)

Appendix: CI using Travis

Build config file = .travis.yml

```
language: r

warnings_are_errors: false
cache: packages

script:
  - R -f run_requirements.R
```