



elixir



Phoenix 1.3 Intro

Ruben Amortegui
@ramortegui
<http://rubenamortegui.com>
<https://github.com/ramortegui>

Agenda

- About Phoenix
- Http Request / Response
- Default Structure and files
- How to create basic pages from scratch
- Generators
- Context
- Important phoenix's topics
- Summary

About Phoenix

- First commit Dec-2013[1]
 - Objectives of the web framework[2]:
 - Performance
 - Productivity
 - Fault tolerance
 - Distributed apps
 - Real Time apps

[1] https://groups.google.com/forum/#!topic/phoenix-talk/l8klhc_LC7o

[2] <https://www.youtube.com/watch?v=s0RxjHs0cwk>

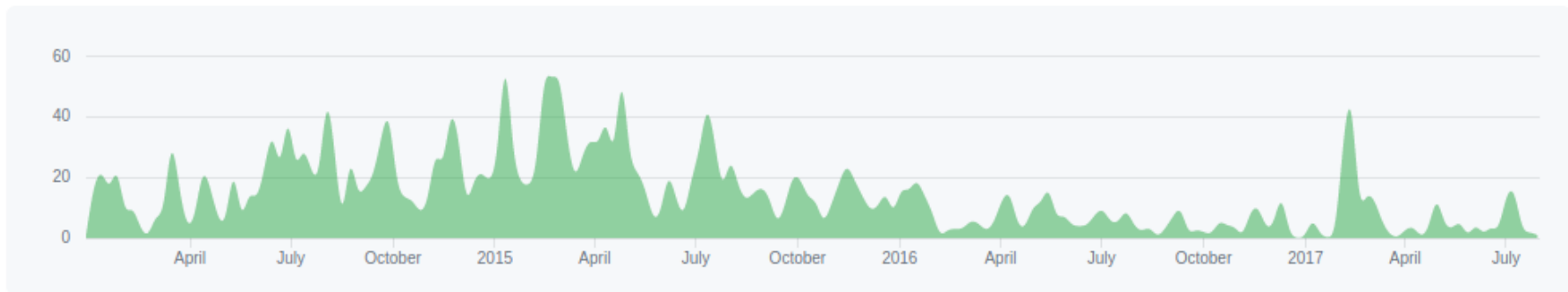
Development

Development[1]

Jan 12, 2014 – Aug 15, 2017

Contributions to master, excluding merge commits

Contributions: **Commits** ▼



Productive. Reliable. Fast. <http://www.phoenixframework.org>

📄 3,887 commits

🔗 9 branches

📦 61 releases

👤 409 contributors

🏠 MIT

[1] <https://github.com/phoenixframework/phoenix>

HTTP Request/Response [1]

- HTTP Request
 - Cowboy (Plug adapter)
 - Phoenix Endpoint
 - Phoenix Router
 - Phoenix Controller
 - Phoenix Views
 - Phoenix Template
- HTTP Response

[1]<https://elixir-examples.github.io/examples/phoenix-framework-from-http-request-to-response>

Default Structure and Files

- mix phx.new hello (live coding)
 - assets (css, js, brunch)
 - _build (compiled files)
 - config (environments)
 - deps (libraries of dependencies of the app)
 - lib (the code)
 - mix.exs (configuration of the application)
 - mix.lock (deps versions, and configurations)
 - priv (i18n files, DB migrations, assets compiled)
 - README.md (Information about the project)
 - test (Test)

Create Page

- <http://localhost:4000/hello>



(live

Create Page using params

- <http://localhost:4000/hello/Ruben>



(live

Generators

- `mix help | grep phx` (outside a phoenix project)

```
ramortegui@RA testing $ mix help | grep phx
mix local.phx          # Updates the Phoenix project generator locally
mix phx.new            # Creates a new Phoenix v1.3.0 application
mix phx.new.ecto       # Creates a new Ecto project within an umbrella project
mix phx.new.web        # Creates a new Phoenix web project within an umbrella project
```

- `mix help | grep phx` (inside a phoenix project)

```
mix local.phx          # Updates the Phoenix project generator locally
mix phx.digest         # Digests and compresses static files
mix phx.digest.clean   # Removes old versions of static assets.
mix phx.gen.channel    # Generates a Phoenix channel
mix phx.gen.context    # Generates a context with functions around an Ecto schema
mix phx.gen.embedded   # Generates an embedded Ecto schema file
mix phx.gen.html       # Generates controller, views, and context for an HTML resource
mix phx.gen.json       # Generates controller, views, and context for a JSON resource
mix phx.gen.presence   # Generates a Presence tracker
mix phx.gen.schema     # Generates an Ecto schema and migration file
mix phx.gen.secret     # Generates a secret
mix phx.new            # Creates a new Phoenix v1.3.0 application
mix phx.new.ecto       # Creates a new Ecto project within an umbrella project
mix phx.new.web        # Creates a new Phoenix web project within an umbrella project
mix phx.routes         # Prints all routes
mix phx.server         # Starts applications and their servers
```

Generators – phx.gen.html

```
mix phx.gen.html Users User users name:string
```

Context

- Thinking about design

Others

- Tests (ExUnit)
- Deployments (server, heroku)
- Brunch (Assets management)
- Channels (bidirectional communication / PubSub)
- Ecto (DB abstraction)
- GraphQL (Absinthe)
- ...

Summary

- HTTP Request/Response is based on transformation of the structure of `%Plug.Conn{...}`
- Generators are good as learning tools.
- Phoenix could be use to serve basic and/or complex web apps.

References

- <https://hexdocs.pm/phoenix/overview.html>
- Halvorsen, Lance (2017). functional Web Development with Elixir, OTP, and Phoenix (Beta). Ebook. The pragmatic Bookshelf.

Thanks!

Q & A?

@ramortegui

<http://rubenamortegui.com>

<https://github.com/ramortegui>