Phoenix Guide

Once Phonenix has been installed successfully, you can follow the following chain (regardless of the operating system you are on.)

View the complete source code along with changes (in commits) here: TODO

Main Git repo URL:

Server running

See: https://github.com/recluze/elixir-phoenix-hello/commit/87bfda0fb484edabc9e70bf551c34323a9d5a3a3 (Changes made by mix.)

Directory structure:

hello: contains business logic hello_web: Contains web-related stuff. Not everything is web related, some might be for mobile etc.

Add a new page:

See official documentation about this here: https://hexdocs.pm/phoenix/adding_pages.html#content

See: https://github.com/recluze/elixir-phoenix-hello/commit/04dd24b518fd509566761c364e2d879ae0c1c4f9

Add a param to page:

See: https://github.com/recluze/elixir-phoenix-hello/commit/bcc96c57842fd3a410eafa328464de9aa68ab8a6

Run:

```
mix phx.routes
```

Take a loo at layout page: `hello_web/templates/layout/app.html.eex`

Add a resource users route

See official documentation about this here: https://hexdocs.pm/phoenix/routing.html#examining-routes

See: https://github.com/recluze/elixir-phoenix-hello/commit/1bdd47dd4ba318c3c11f18df35ab7647f59bda68

```
mix phx.routes
```

(We will have to create new controller etc.)

Path helpers

See official documentation about this here: https://hexdocs.pm/phoenix/routing.html#path-helpers

See: https://github.com/recluze/elixir-phoenix-hello/commit/20f3994416bf4ad01082758e1d500f74ba23bac8

Remove route

See: https://github.com/recluze/elixir-phoenix-hello/commit/4b1dfb4634b55a93c7e3e9dc15c801e12ce3e9b1

Flash message

See: https://github.com/recluze/elixir-phoenix-hello/commit/804da956482db0ba106ae07e4a501351919134e6
See flash location in: lib/hello_web/templates/layout/app.html.eex

Variables to Templates

See: https://github.com/recluze/elixir-phoenix-hello/commit/9566215d8e605e22527045c8aed4a09b2353b047

Manually setting content type

See: https://github.com/recluze/elixir-phoenix-hello/commit/6e353ca8eb7626d4b11c597c8fd27e3a35cd218f

Rever last

See: https://github.com/recluze/elixir-phoenix-hello/commit/a5e9c2210e9e3a78b93d4264976713de23dec8e6

Calling Functions from Views

See: https://github.com/recluze/elixir-phoenix-hello/commit/541cf4b804f4476b1fea8cbb1d338c17fd5567a7

Channels

See official documentation about this here: https://hexdocs.pm/phoenix/channels.html

- Define a new channel through lib/hello_web/channels/user_socket.ex
- Implemenet a channel handler in lib/hello_web/channels/room_channel.ex

See: https://github.com/recluze/elixir-phoenix-hello/commit/a5572b965f242cd181ee32a4d7dca1dcaa5cf643
Also: https://github.com/recluze/elixir-phoenix-hello/commit/9b6790927bcb550d963bd335f0888b6a6fe5302f

- Open http://localhost:4000
- Add message sending and broadcasting See: https://github.com/recluze/elixir-phoenix-hello/commit/ed2045333173804e82fe9ff364cc8d5d84e52f0f
- Open two windows on localhost and send message from one to another.
- See elixir log for new messages
- See chat example by Phoenix's lead developer here: https://github.com/chrismccord/phoenix_chat_examp

Databases and Data Persistence through Ecto

See details about Ecto and ORM here: https://hexdocs.pm/phoenix/ecto.html#content

Create the first model

```
mix phx.gen.schema User users name:string email:string bio:string number_of_pets:integer
```

See lib/hello/user.ex and priv/repo/migrations/*_create_user.exs

```
mix ecto.migrate
```

See: https://github.com/recluze/elixir-phoenix-hello/commit/850d0b6791fbf5227513cb9aded5752abb622466 (Auto created by mix)

See: lib/hello/user.ex Also see config/dev.exs

```
iex -S mix
```

```
alias Hello.User
cs = User.changeset(%User{}, %{})
cs.valid?
cs.errors
```

See detailed Ecto talk here: https://www.youtube.com/watch?v=YQxopjai0CU

Make number of pets optional

See: https://github.com/recluze/elixir-phoenix-hello/commit/9baa22670340f00ec58e15d066623ec28f82be2d

```
# Still in the same shell
recompile()
cs = User.changeset(%User{}, %{})
cs.errors    # notice reduced errors

# Let's fix the errors

params = %{name: "Joe Example", email: "joe@example.com", bio: "An example to all", number_c
cs = User.changeset(%User{}, params)
cs.valid?
cs.errors
cs.changes
```

Add minimum length constraint

See: https://github.com/recluze/elixir-phoenix-hello/commit/2e4e06f5716a95d65899bae043fdb30d865d0070

```
# Still in the same shell
recompile()

cs = User.changeset(%User{}, %{bio: "A"})
cs.errors
```

See more constraints here: https://hexdocs.pm/ecto/Ecto.Changeset.html

Using Repo

```
import Ecto.Query
Repo.all(from u in User, select: u.email)

# Nothing has been saved yet

Repo.insert(%User{email: "user1@example.com"})
Repo.insert(%User{email: "user2@example.com"})

Repo.all(from u in User, select: u.email)

# No validation was performed!

Repo.all(from u in User, select: %{u.id => u.email})

Repo.all(User)
# Control+C (twice) to quit
```

You can also use MySQL. Read up more about it here: https://hexdocs.pm/phoenix/ecto.html#using-mysql

Contexts and the CVT Pattern

Controller/View/Template (More commonly known as Model/View/Controller or MVC)

```
mix phx.new foo
cd foo
```

```
mix ecto.create
mix phx.server
```

Add Accounts context

```
mix phx.gen.html Accounts User users name:string username:string:unique
```

Notice the plural in context (Accounts) and singular in module (User).

A file is created as `lib/foo/accounts/user.ex`

Fix the route as before: https://github.com/recluze/elixir-phoenix-foo/commit/d040743df116bbf2962488d12c18e46f22480525

Imprtant file: `lib/foo_web/router.ex`

`mix ecto.migrate`

Kill server then refresh browser.

Go to: http://localhost:4000/users See console for logs. No users are present so none are returned.

Click on 'New User' and then submit without entering any data. See error in "Flash".

Enter proper data and then submit. See user inserted correctly. Hit 'Back' to see list of users.

Take a look at: `lib/foo_web/controllers/user_controller.ex`

Creating Credentials

```
# Do not run migrations after this, just now
```

```
mix phx.gen.context Accounts Credential credentials email:string:unique user_id:references:u
```

Notice "injecting" lines. No HTML scaffolding was created because we used phx.gen.context instead of phx.gen.html

See: https://github.com/recluze/elixir-phoenix-foo/commit/fa73db466282616735a6faa79110f9df41dc749a (auto created)

Change some properties of credentials migration See: https://github.com/recluze/elixir-phoenix-foo/commit/a78d54714db70171eb1e42d30dabc378673c020b

Associations between Users and Credentials

See: https://github.com/recluze/elixir-phoenix-foo/commit/d9efd4ced827724659ac2b30d7569018a0a25899

```
mix ecto.migrate

# In case of errors with fields, you might want to reset the database
# Warning, this will drop all tables and reset all data!
mix ecto.reset
```

Changes to Form

See: https://github.com/recluze/elixir-phoenix-foo/commit/9321aea72c7b2c419ea88e9206e7bb67bbe6a30e

Try inserting new data at: http://localhost:4000/users/new

And you'll notice that email is ignored. This is because we haven't called their changesets yet.

Call changeset of cred on user insertion and deletion See: https://github.com/recluze/elixir-phoenix-foo/commit/37716789525866b9f4e9c1ee43d76c7a24ccafa0

Try inserting new data again: http://localhost:4000/users/new

Show Email in Users List

See: https://github.com/recluze/elixir-phoenix-foo/commit/e664a99235554ebe42a172e15e8b848c7b550bea

Now we need to allow users to login with their email

Create Session Controller and Route

See: https://github.com/recluze/elixir-phoenix-foo/commit/270d31a3eadc32ca72d30099fe231faf1b61f5c4 (correction for session_controller.ex in next commit.)

Add Session View and Controller for Login/Logout

See: https://github.com/recluze/elixir-phoenix-foo/commit/d1b1a2554a1ac4990b101c6cb52d5fbc844c1bba

Create Some Pages to Protect

```
mix phx.gen.html CMS Page pages title:string body:text views:integer --web CMS
```

See: https://github.com/recluze/elixir-phoenix-foo/commit/a049778913257bfc04a3b889278700841128a048 (Manual changes in router.ex)

Correction: See: https://github.com/recluze/elixir-phoenix-foo/commit/e8433102843a88d329de2c0cf0630cb3b14

```
mix ecto.migrate
```

See: https://github.com/recluze/elixir-phoenix-foo/commit/cc9b986d383483a7e3402fdd136144be40f5f614 (Hide 'view' field from form) Also: https://github.com/recluze/elixir-phoenix-foo/commit/d07c1696e1f62389249dfd644c4650db410c9831 (fix views default) Also: https://github.com/recluze/elixir-phoenix-foo/commit/f73787c5b5d9c85df2b40feaa5da0cc326699303 (Remove views constraint from cast)

Open browser at: http://localhost:4000/cms/pages Gives error: login required

Goto: http://localhost:4000/users/ and find an email to login with. Than: http://localhost:4000/sessions/new and login Finally: http://localhost:4000/cms/pages

Let's go back and view the session controller.

Create Authors for Creating Pages

These are different from our "users" since not all users are authors.

```
mix phx.gen.context CMS Author authors bio:text role:string genre:string user_id:references:
```

(Automatically Created) See: https://github.com/recluze/elixir-phoenix-foo/commit/707fd8c275c167cf2aafe7d5a7fea773c23a9c57

Integrity constraint for Authors: See: https://github.com/recluze/elixir-phoenix-foo/commit/fb08191e11031bdc5535c2f4e1d92c903875cb2a

Add Author Association to Pages

```
mix ecto.gen.migration add_author_id_to_pages
```

(auto) See: https://github.com/recluze/elixir-phoenix-foo/commit/fb08191e11031bdc5535c2f4e1d92c903875cb2a (with previous)

See: https://github.com/recluze/elixir-phoenix-foo/commit/389f82485dd92faf4cb3453bf8d313322a21dcd4 Typo Fix: https://github.com/recluze/elixir-phoenix-foo/commit/f6cbf6076bdedc75904abdd9f3f3f9ad04ddce25 (typo)

```
mix ecto.migrate
```

Fix schemas so that pages belong to authors See: https://github.com/recluze/elixir-phoenix-foo/commit/f3ce62560f0bf089ee293ca7719b5846d80540e5

Preload authors when loading pages

See: https://github.com/recluze/elixir-phoenix-foo/commit/0d293a2af86184aa0eaa256c97469f2cd2ac7eca

Save authors when saving pages

See: https://github.com/recluze/elixir-phoenix-foo/commit/6f10912e68e540c4c457dc11dadfd1347d21e597

When viewing pages, ensure correct author

See: https://github.com/recluze/elixir-phoenix-foo/commit/cb74e18bc2d2eb7518d0de66f03f48f2ec0b18ab (incorrect file edited) Fix: https://github.com/recluze/elixir-phoenix-foo/commit/cbfcfb854a4c78ed2b9873ed71ab1271de8fa37c

Go back to https://github.com/recluze/elixir-phoenix-foo/commit/6f10912 and add ensure author exists now.

Edit, Create of Page Should Handle Proper Author

See: https://github.com/recluze/elixir-phoenix-foo/commit/8df90ffebc0bedccb6e27575ebd70f65c6d21576

Go to http://localhost:4000/sessions/new and login Then, go to http://localhost:4000/cms/pages and create a new page

Adding logging to see what's going on

Just go to foo/cms/cms.ex and add an IO.inspect at the last line of create_page. Insert new page and see output on console.

Adding View Count

Recall that we left views at 0 and hid them from the frontend

See: https://github.com/recluze/elixir-phoenix-foo/commit/b78a1faaa84d4169df565cbf6e9876d7c768b993

Go to: http://localhost:4000/cms/pages/1 Refresh a couple of times to see count increment. Also see log.

See detailed Ecto talk here: https://www.youtube.com/watch?v=YQxopjai0CU