**Making websites with Actix Web**

First, make sure you have rustup installed.

**Make a new project:**

cargo new foo\_app

**Run/Compile your project:**

Navigate to your project’s root directory (using cd), then use:

cargo run

Then open your browser and navigate to <http://localhost:8080/> (unless you specified a different port number than 8080).

**Printing out a page of uninteresting HTML:**

main.rs:

use actix\_web::{error, middleware, web, App, Error, HttpResponse, HttpServer};

async fn index() -> Result<HttpResponse, Error>

{

Ok(HttpResponse::Ok().content\_type("text/html").body("<h1>Hello, World!</h1"))

}

#[actix\_rt::main]

async fn main() -> std::io::Result<()>

{

HttpServer::new(|| {

App::new()

.route("/", web::get().to(index))

})

.bind("127.0.0.1:8080")?

.run()

.await

}

**A basic social media app:**

[Beginning from a fresh rust app, ignore Printing basic HTML”] Open up your Cargo.toml file and add some dependencies.

[dependencies]

actix\_web = “2.0.0”

actix\_files = “0.2.1”

You’ll need actix\_files to incorporate things like a favicon or static files. As other dependencies need to be added, put them in this same list in the Cargo.toml file. Now, open up main.rs in your src folder and delete the cute little Hello World in it. Instead, replace it with:

#[macro\_use]

extern crate actix\_web;

//If you want to have static content:

use actix\_files as fs;

mod templates; //This will be created and explained shortly

…Add the favicon, all page functions, and of course create the app in the main() function

Up to this point, your main.rs file should look like this.

Now, create a new file called templates.rs and open it. It can be in src. This is where our Markup templates will go, and they describe the HTML that the app will be sending to the users. I prefer to put the template code in a separate file just so that the main.rs file doesn’t begin to feel so overwhelming and complex to read through, but you can absolutely put everything in this file into main.rs (if you do, get rid of the mod templates; line).