CXP Elixir Codebase Setup Guide:

Clone Elixir Codebase Repo using this [link](https://gitlab.com/velocity-global/velocity)**Local Source Setup: (Mac/Linux)**

1. I**nstall ASDF and add elixir plugin:**

* <https://asdf-vm.com/#/core-manage-asdf-vm>
* asdf plugin add erlang or [download using this](https://www.erlang.org/downloads/23.1)
* asdf plugin add elixir or [download precompiled version](https://github.com/elixir-lang/elixir/releases/download/v1.11.0/Precompiled.zip)
* Install Postgres DB v13 using this [link](https://www.postgresql.org/download/)

2. **Run these commands to get the dependencies set up:**

* asdf install
* mix local.hex
* mix archive.install hex phx\_new 1.5.9
* mix deps.get

3. **If this is your first time starting the API, you will also need to create/seed a local database:**

* Configure your local database settings in config [velocity/config/dev.exs]
* mix ecto.setup

4. **install and start redis (caching engine):**

* brew install redis
* brew services start redis

5. **Start the phoenix server:**

* mix phx.server

6. S**cript:**

* mix setup [install dependency and db migration]
* mix ecto.reset [reset database]
* mix test [run tests]

7. **Imp. Files:**

* velocity/mix.exs [Project Dependency configuration file]
* velocity/priv/repo/seeds.exs [Db pre seed data file]
* velocity/lib/velocity\_web/router.ex [routing and endpoint configuration file]
* velocity/config [dev and prod config management folder]
* velocity/cicd [cicd scripts folder]

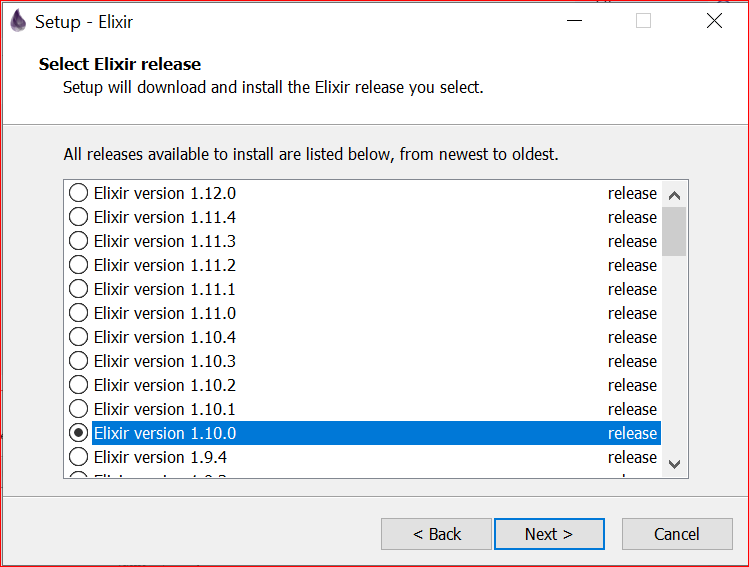
**Local Source Setup: Windows**

1. Prerequisites

Erlang 23.1 - <https://www.erlang.org/downloads/23.1>

Elixir 1.10 - [https://elixir-lang.org/install.html#windows](https://elixir-lang.org/install.html#windows )

* 1. Choose the version (1.10.0) while installing



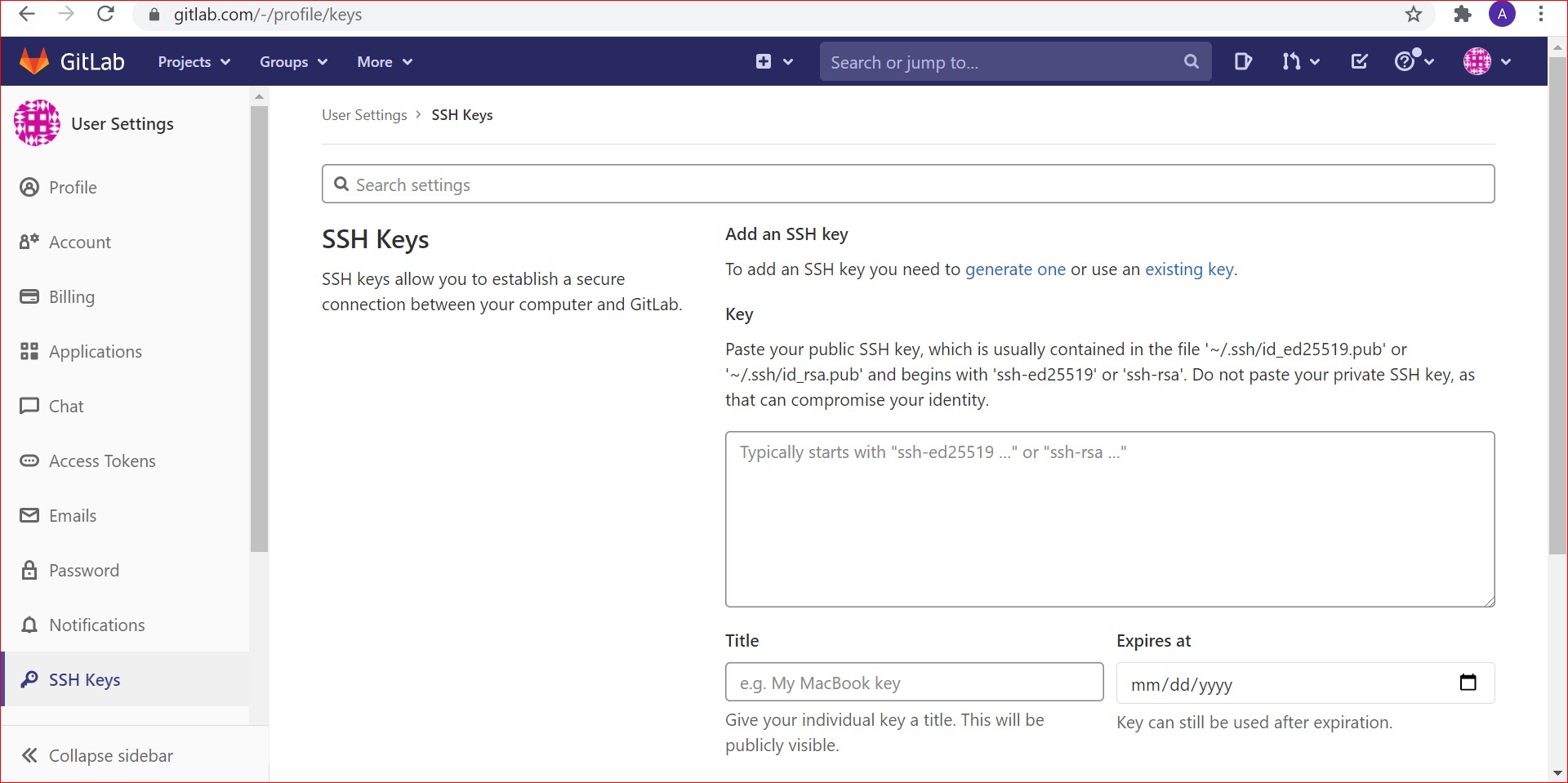
Redis 2.4.5 - <https://github.com/dmajkic/redis/downloads>

postgresql - <https://www.postgresql.org/download/> - PostgreSQL download v-13.3

* 1. Follow the steps in this link for installation <https://www.guru99.com/download-install-postgresql.html>

**Steps to generate SSH:**

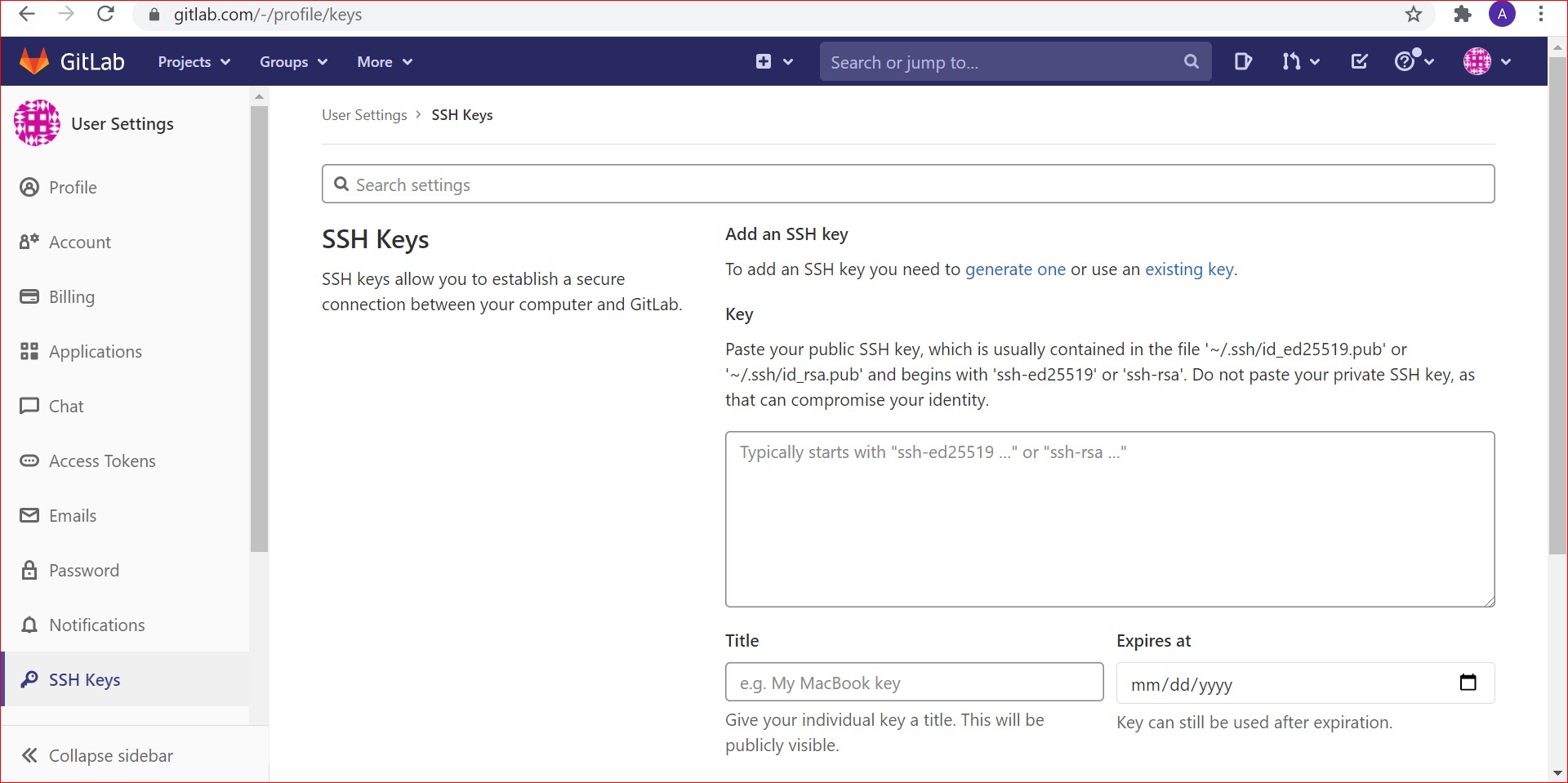
1. Click on Add SSH Key link in Git Lab
2. Tap on generate one link in the below screen



1. In Command prompt give the below command
   1. ssh-keygen -t rsa -b 2048 -C "your email ID"
   2. Tap on Enter key Enter passphrase
   3. Your public key has been saved in your local path Eg: C:\Users\anjana.angappan/.ssh/id\_rsa.pub
   4. Copy the public key



3.5 Paste the SSH key in Gitlab text area and tap on Add key



**Steps to clone:**

1. Add required SSH settings on Gitlab

Use below links to clone the repo using different tools **(Recommended is Tortoise Git)**

Github desktop - <https://learn.hibbittsdesign.org/gitlab-githubdesktop/cloning-a-gitlab-repo>

Visual Studio Code - <https://docs.microsoft.com/en-us/azure/developer/javascript/how-to/with-visual-studio-code/clone>

Tortoise Git - <https://tortoisegit.org/docs/tortoisegit/tgit-dug-clone.html>

1. Download Tortoise Git
2. Select OpenSSH and install Tortoise Git
3. Click on “Recommend for windows” option
4. Download and install Git
5. Select the path where the source code need to be cloned and create a new folder
6. Right click on the folder and Choose tortoise git
7. Paste the clone URL
8. Click on Yes
9. Code will be get cloned successfully in the folder

2. **Run these commands to get the dependencies set up in project root directory**

* Move to Project folder/CMD
* Give the below command
* mix local.hex
* mix archive.install hex phx\_new 1.5.9
* mix deps.get

**Note:** You will get timeout out while running the command **mix deps.get.** Now give the command (For Windows) mentioned under **Common Error during Setup.**

* mix deps.get - Give this command again after completing the above step.

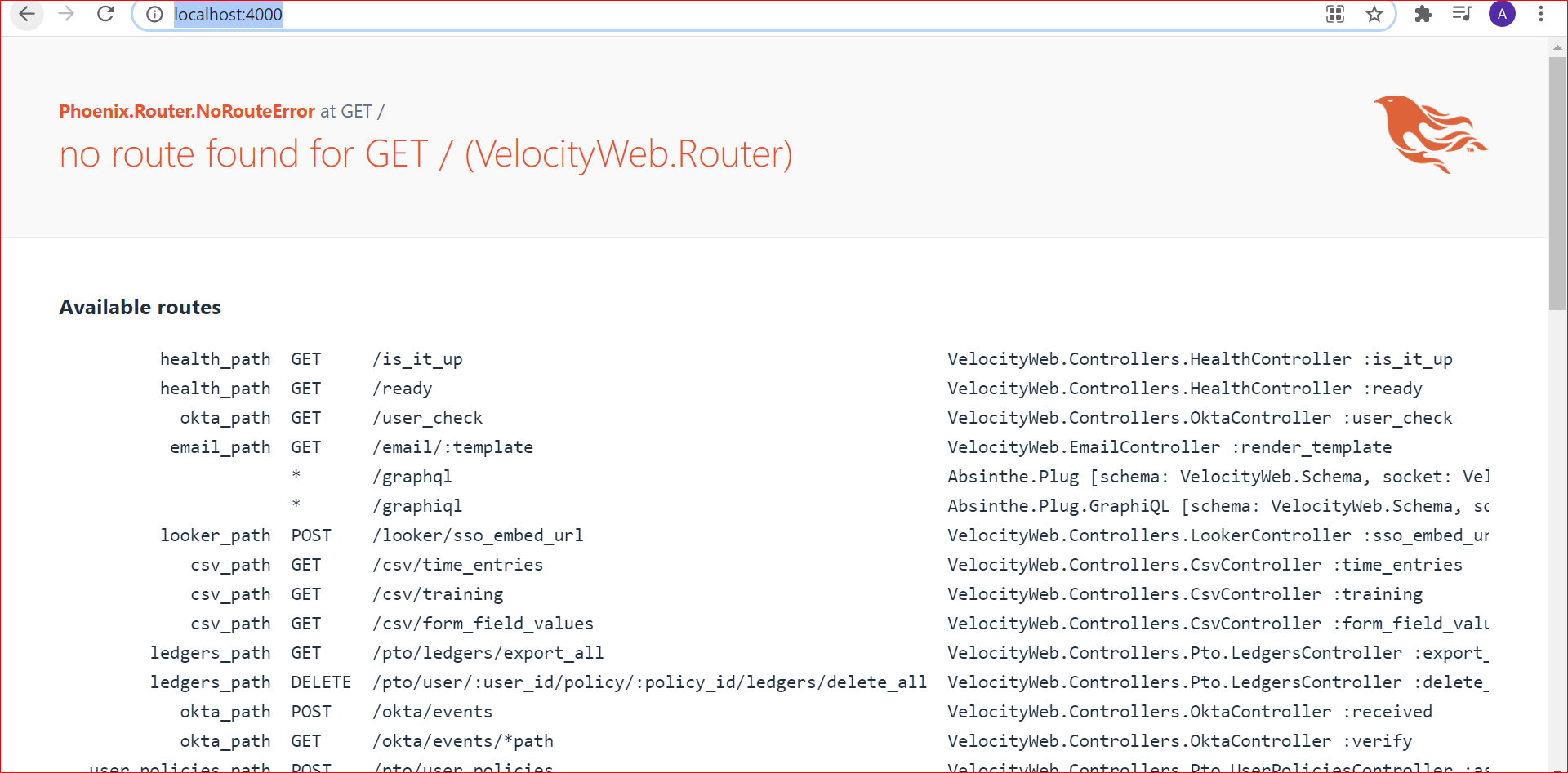
3. **If this is your first time starting the API, you will also need to create/seed a local database:**

* mix ecto.setup
* Open Reddis folder - 64bit and click on redis-server.exe (Reddis should running)

4. **Start the phoenix server:**

* mix phx.server

5. <http://localhost:4000/> - Give this in browser and ensure whether the backend application is installed



### **Docker:**

If this is your first time starting the API, you will also need to create/seed a local database:

* docker-compose run web mix ecto.setup
* docker-compose up
* docker-compose up -d db

**Common Error during Setup:**

1. Timeout Error:

* Error Statement:  
  Request failed (:timeout)  
   If this happens consistently, adjust your concurrency and timeout settings:

HEX\_HTTP\_CONCURRENCY=1 HEX\_HTTP\_TIMEOUT=120 mix deps.get  
 \*\* (Mix) Package fetch failed and no cached copy available (<https://repo.hex.pm/tarballs/absinthe-1.6.3.tar>)

* Solution:
* For Unix and Linux:
  + HEX\_HTTP\_CONCURRENCY=1 HEX\_HTTP\_TIMEOUT=120 mix deps.get
* For Windows:
  + mix hex.config http\_concurrency 1
  + mix hex.config http\_timeout 120

**React App Setup**

Clone Velocity Global Customer Management UI - React App using this [link](https://gitlab.com/velocity-global/velocity-ui)

**Prerequisites**

Node JS >=14 required <https://nodejs.org/en/download/>

- [Yarn](<https://classic.yarnpkg.com/en/docs/install/>) and [Node](<https://nodejs.org/en/download/>) are installed on your machine. (Note that in most cases, Yarn installs Node as a dependency)

**npm install --global yarn**

Make sure that the .env file is located on the root folder of the project(there is no .env file on the given repo, Feel free to ask me [@Jothikannan Chandramohan](mailto:jothikannan.chandramohan@Impigertech.com)

**Running the APP**

Run the following comments

**- yarn install**

**- yarn start**

A server will be running at <http://localhost:8080> - Give this in browser and ensure whether the frontend application is installed

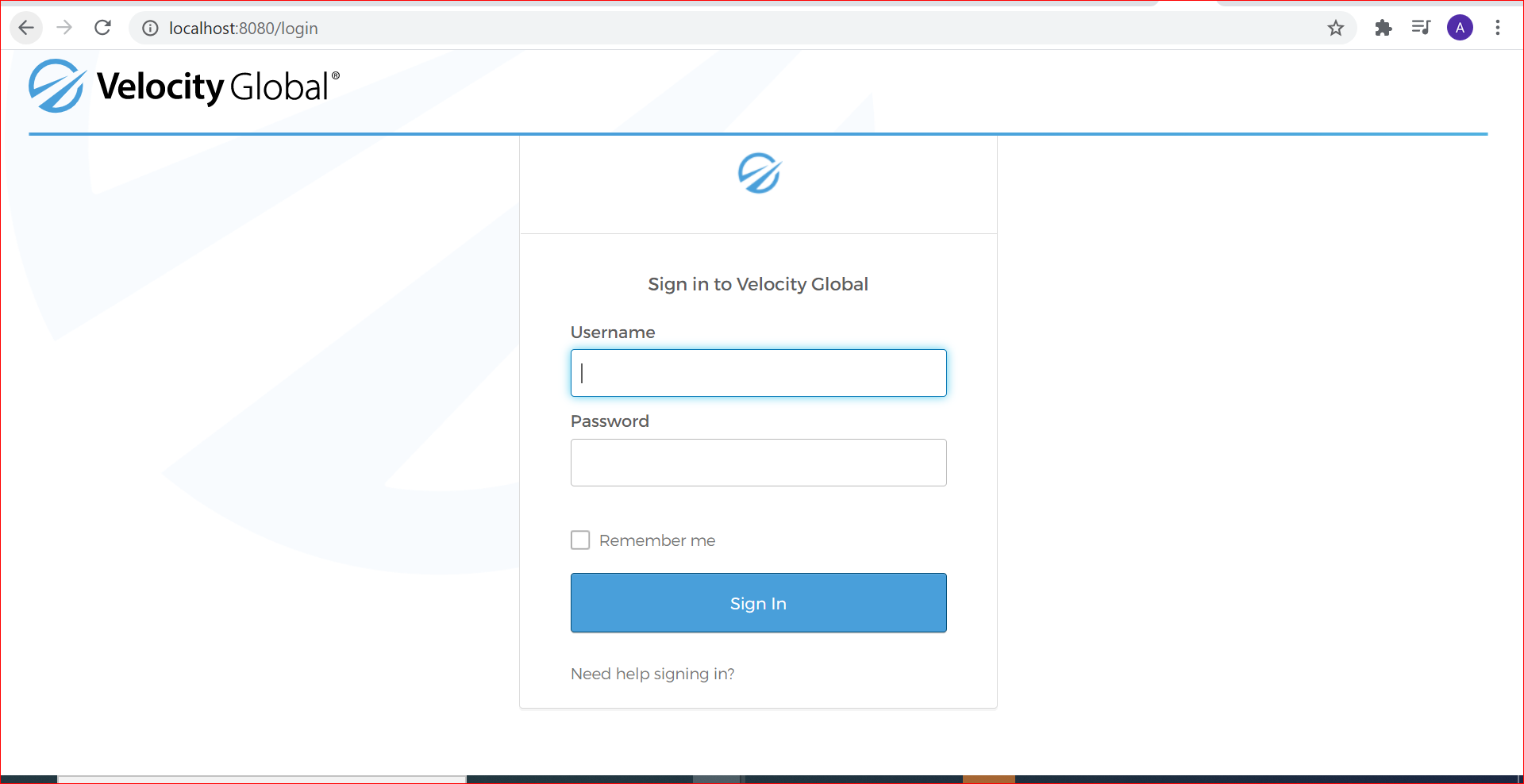
For the Okta bypass changes, Need to do following changes in Front end code

Goto src/shared/config/oktaConfig.js

Line number 12

scopes: ['openid', 'profile', 'email', ‘groups’] - Change this into

scopes: ['openid', 'profile', 'email']

-

**To run the application(**http://localhost:8080) **ensure the following steps are done:**

1. **Run the Reddis –** Move to Reddis folder **->** 64bit -> click on Reddis-server.exe
2. Open the folder where you have cloned the backup code and run CMD
3. mix phx.server - Give this command in CMD
4. Open the folder where you have cloned the frontend code and run CMD
5. yarn start - Give this command in CMD
6. Ensure <http://localhost:8080> application is up

**To Clone the dev changes follow the below steps:**

1. Create a new folder in your local (Eg: DevBranch)
2. <https://gitlab.com/velocity-global/velocity-ui/-/tree/dev-myguides> - Clone this Branch
3. Copy the env file and paste into this folder
4. Open the folder and run CMD
5. yarn install
6. yarn start

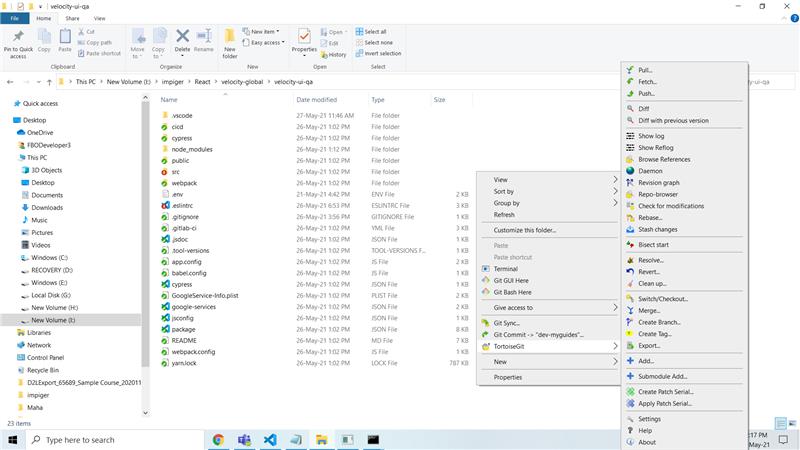
**Do the following changes in backend code for testing:**

1. Replace the **permissions\_seeds.ex** in this path (velocity/lib/velocity/Seeds/permissions\_seeds.ex) - File is attached in the ticket <https://impiger.atlassian.net/browse/IWG-2906>
2. Replace the **config.exs** in this path (velocity/config.exs) - File is attached in the ticket <https://impiger.atlassian.net/browse/IWG-2906>
3. Open the folder where you have cloned the backup code and run CMD and give the below commands in sequence.
4. mix ecto.drop
5. mix ecto.setup
6. mix phx.server

Start testing

**Steps to pull the code with any new changes is pushed:**

1. Open the folder in ur local machine (Eg: DevBranch)
2. Right click -> Tortoise -> Pull



1. Now reload the application http://localhost:8080/