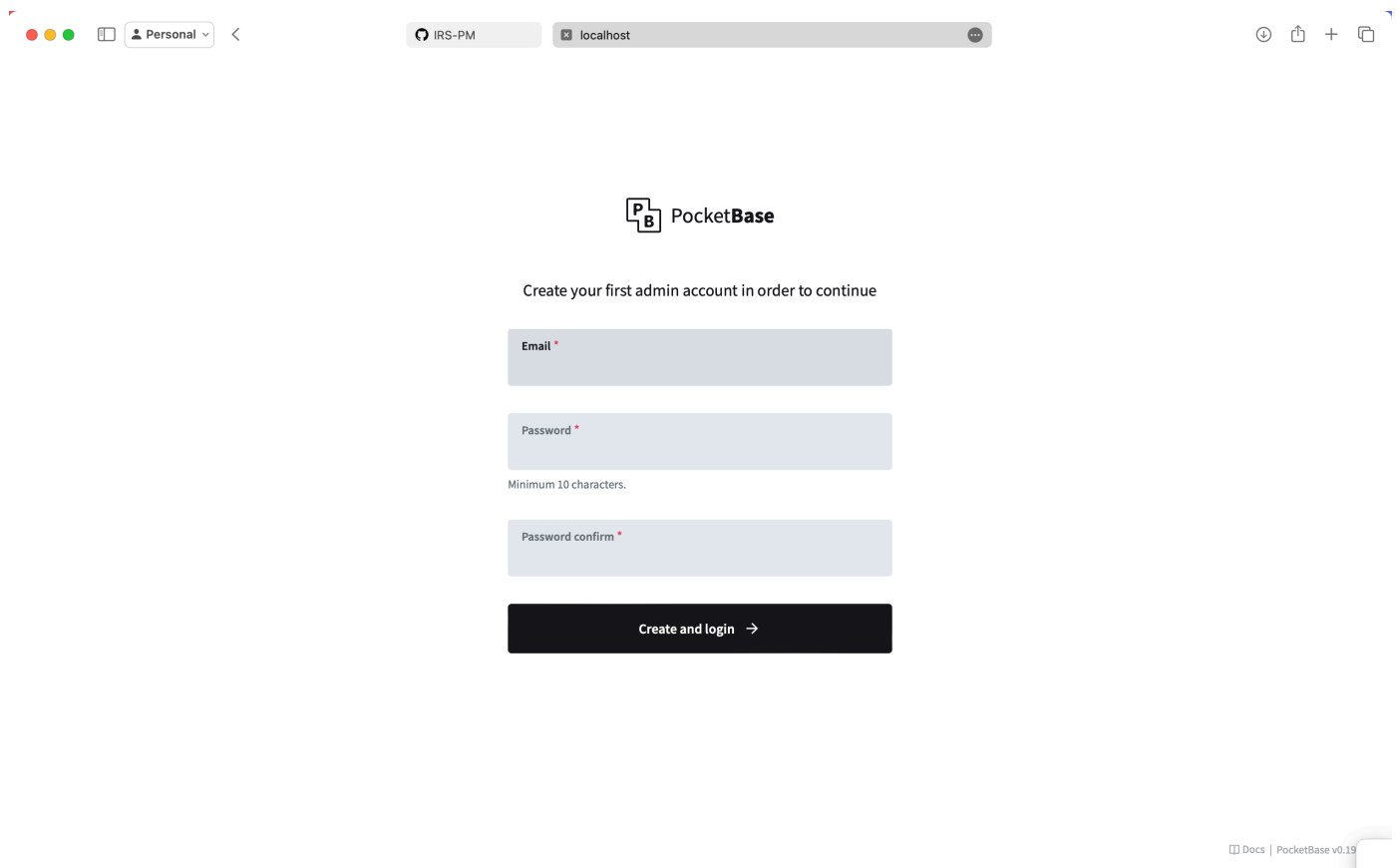# Mango - User Guide

## Prerequisites

- **Docker**
- **Internet access:** Pre-trained paper embeddings are stored on *Managed Milvus Cloud*

## User Database Setup

Move to the same folder where `docker-compose.yaml` file are placed. Run the following command to set up your user profile database -

```
docker-compose up -d
```

Then, open webpage `localhost:8090/_/` to sign up your admin account -

After you logged into the pocketable, move to the "settings" page and import the table schema with user data -

- Import `mango-user/pb_schema.json` to create collections.
- Upload `mango-user/backup1.zip` into the backup page and restore.

> *The container may shutdown but it is OK. Don't worry.*

Run `docker-compose down` and re-run `docker-compose up -d` to restart the container. Then heading to the admin page you will find the records are already restored.
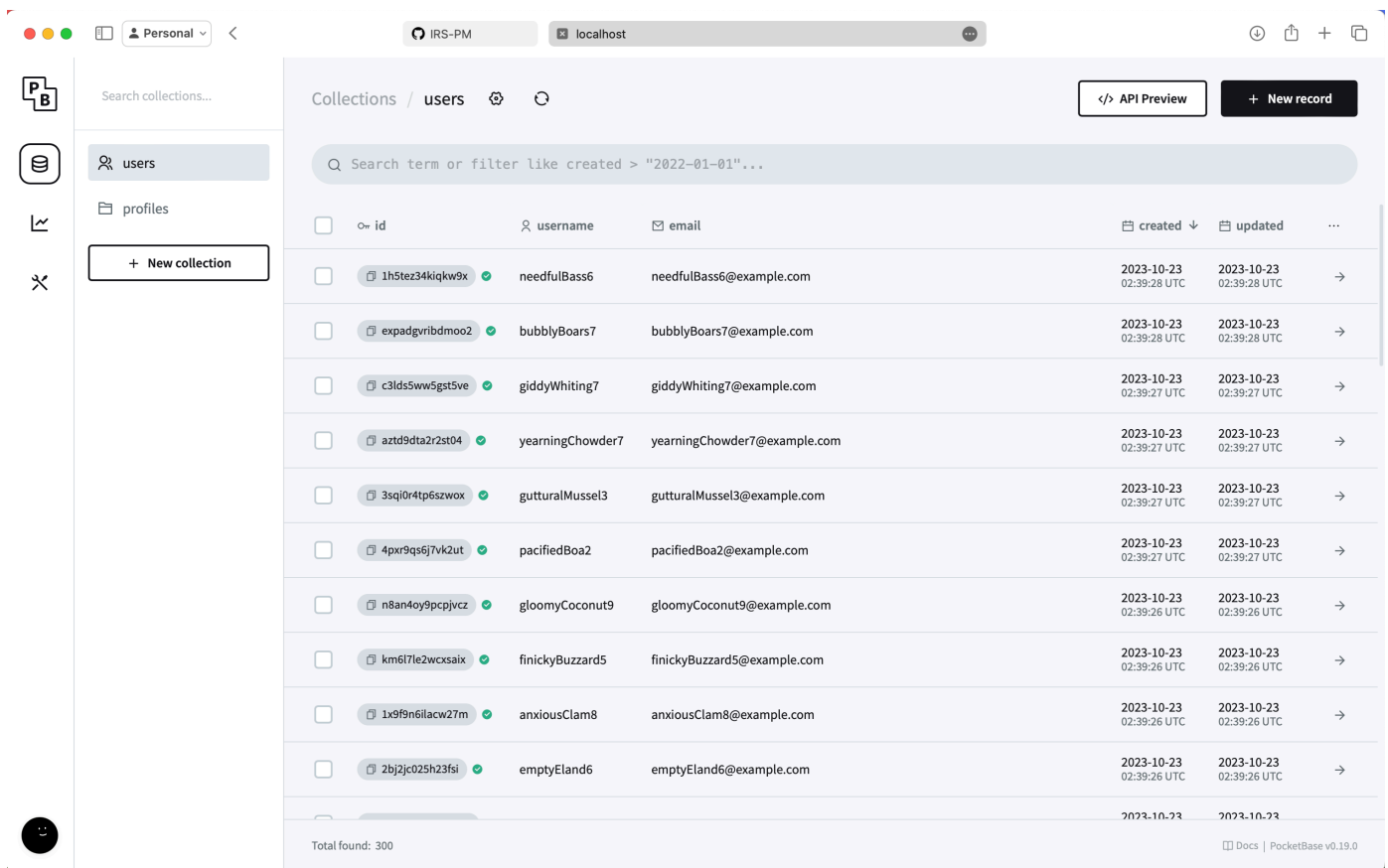


Then, you have successfuly setup the user database.

# Backend Setup

Go to the `Mango-server` folder, and download the models, stores, and config file from our Google Drive. Unzip the `zip` file after successfully downloaded it.

- Download Link: https://drive.google.com/file/d/1M2TNcVArPhBpWrU9-Ok4fNLn3LXj-u2B/view?usp=share_link

First, please follow official guide to install **Poetry** -

- https://python-poetry.org/docs/#installing-with-the-official-installer

Then run the command to install dependencies and setup Mango backend server -

```
# Install dependencies
poetry install

# Setup mango server
uvicorn main:app --reload --host 0.0.0.0
```

Once the server has been successfully activated, you will see logs like this -
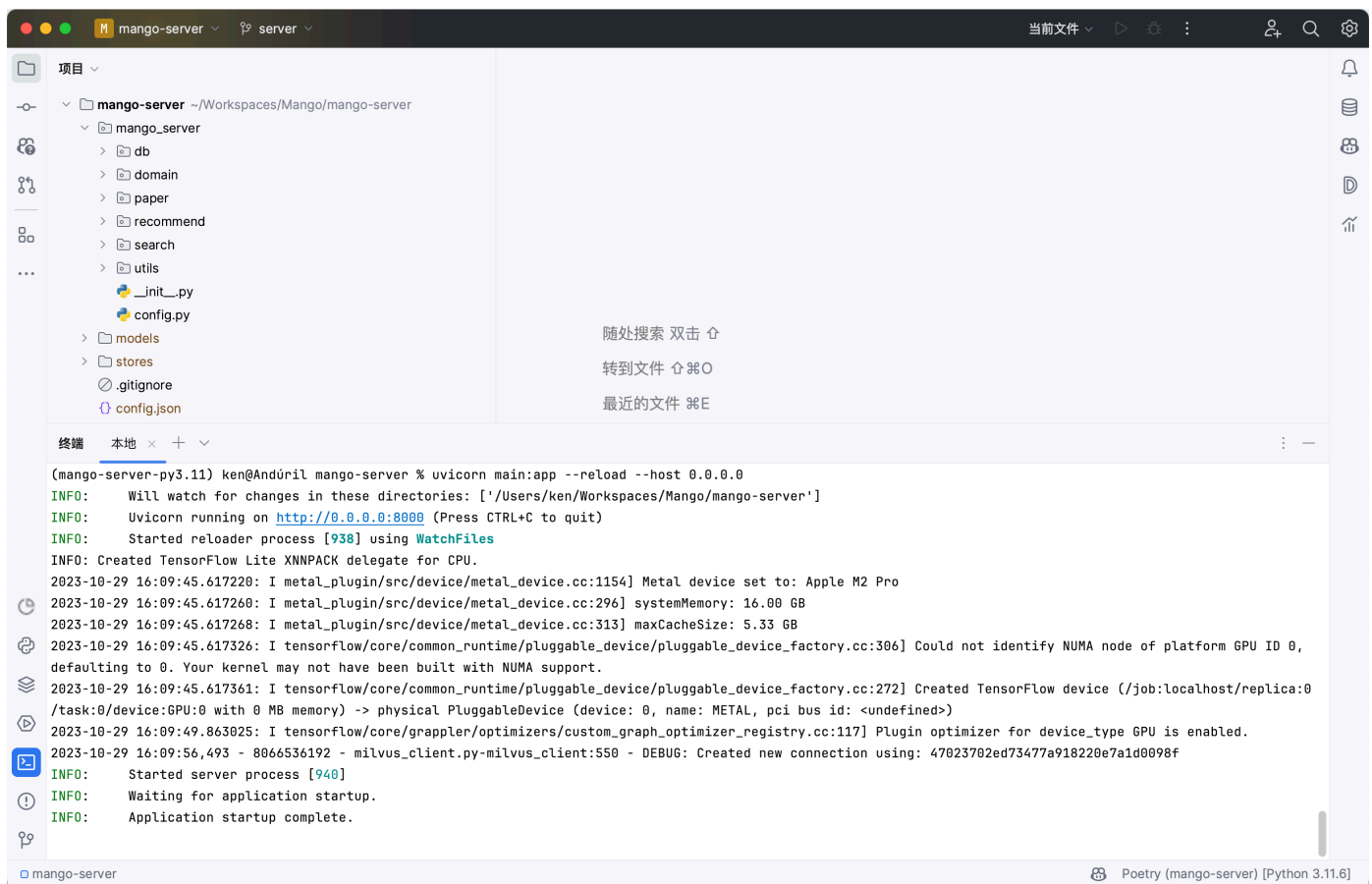


# Frontend Setup

First, please install `node` or `bun` on your device.

Then go to the `Mango-web` folder install all the dependencies of the project with `npm install .` or `bun install .` . Once installed all the dependencies, run the commend to setup mango Web server for the UI -

```
# If using npm
```

```
npm run dev

# If using bun
bun --bun run dev

# ken@Andúril Mango-Web % bun --bun run dev
$ vite dev
▲ [WARNING] Cannot find base config file "./.svelte-
kit/tsconfig.json" [tsconfig.json]

    tsconfig.json:2:12:
      2 │    "extends": "./.svelte-kit/tsconfig.json",
        │                ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~


Forced re-optimization of dependencies

  VITE v4.4.11  ready in 843 ms

  →  Local:    http://localhost:5173/
  →  Network: use --host to expose
  →  press h to show help
```

Then, open the local url in your browser, you will see the UI shown below and you are good to go!

## Conformal symplectic and relativistic optimization

**Neural Information Processing Systems**   2019

G. Francca   Jeremias Sulam   Daniel P. Robinson   R. Vidal

TLDR;   This work proposes a new algorithm based on a dissipative relativistic system that normalizes the momentum and may result in more stable/faster optimization, and generalizes both Nesterov and heavy ball.

**READ! →**

## Compositional Generalization in Image Captioning

**Conference on Computational Natural Language Learning**   2019

Mitja Nikolaus   Mostafa Abdou   Matthew Lamm   Rahul Aralikatte   Desmond Elliott

TLDR;   A multi-task model is proposed that combines caption generation and image–sentence ranking, and uses a decoding mechanism that re-ranks the captions according their similarity to the image, which is substantially better at generalizing to unseen combinations of concepts compared to state-of-the-art captioning models.

**READ! →**