

# Fullstack Development

# Preflight project - frontend

[Github Repo](#)

# Frontend framework

Tool that enables you to develop the frontend of your applications

## Why?

- You don't have to everything code from scratch.
- You can collaborate with others more easily.

# Frontend JS framework

- [State of JS 2023](#)
- [Stack Overflow Survey 2023](#)

# Frontend tooling

- Code bundler ("*build tools*")
- Development server
  - Hot module replacement (HMR)

# Frontend tooling

- Vite
- Webpack
- Parcel
- Rollup

# Setup

- React
  - `npm create vite@latest` (*Follow the instruction*)
- Additional libraries
  - `npm i dayjs axios @picocss/pico`

# Files

- Remove all `css` files
- `./vite.config.ts` ([Link](#))
- `./src/main.ts` ([Link](#))
- Modify `./package.json` ([Link](#))
- `./src/types.ts` ([Link](#))
  - I use [Quicktype](#) to generate this.



# Minimal example

- `./src/App.ts` [\(Link\)](#)
- Start dev
  - `npm run dev`

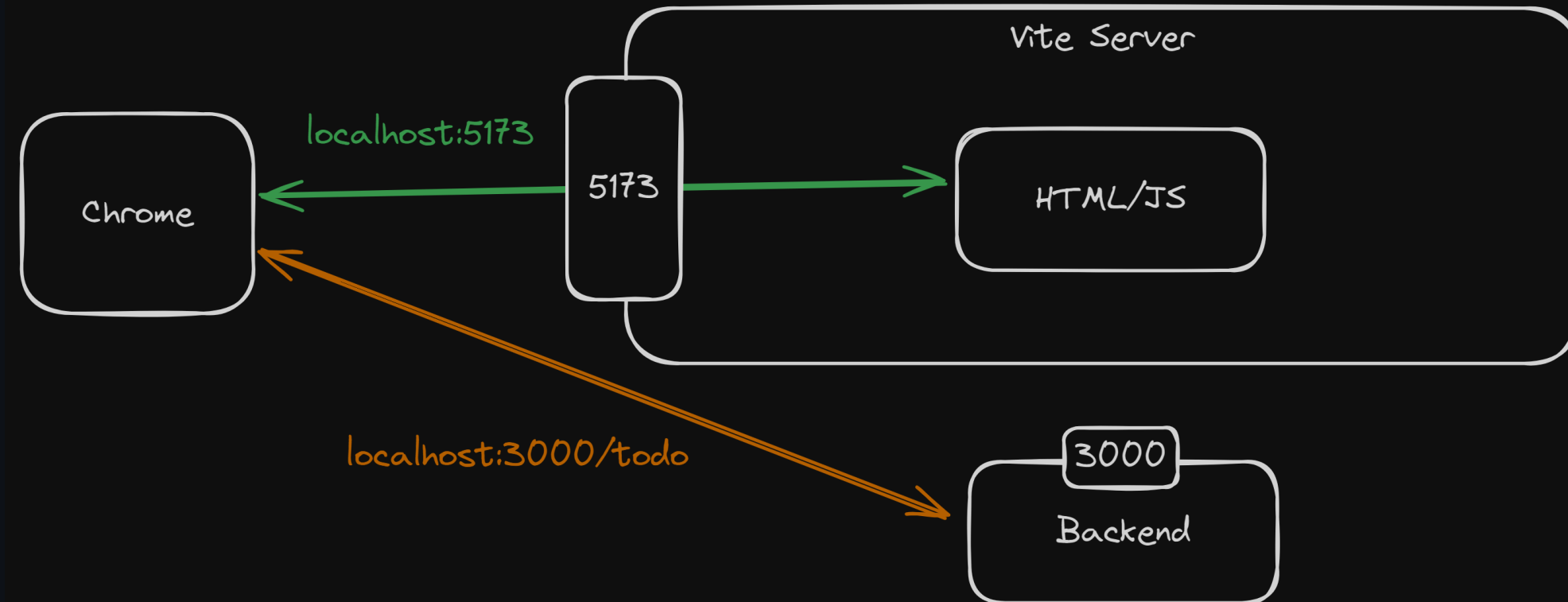
# Full example

- `./src/App.ts` [\(Link\)](#)
- Build
  - `npm run build`
- Preview
  - `npm run preview`

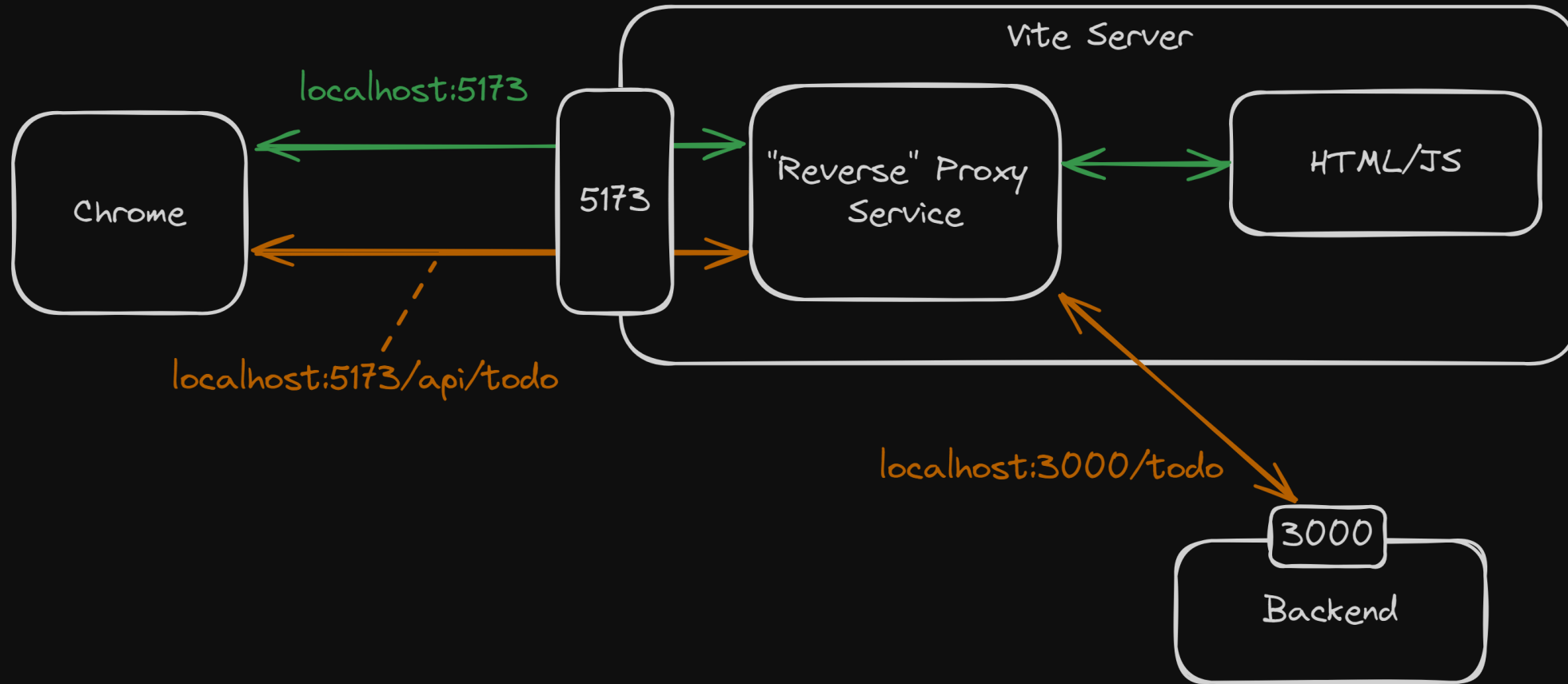
# Why reverse proxy?

- No need to expose your backend to the internet.
- No issues with CORS and cookies.







# Without reverse proxy

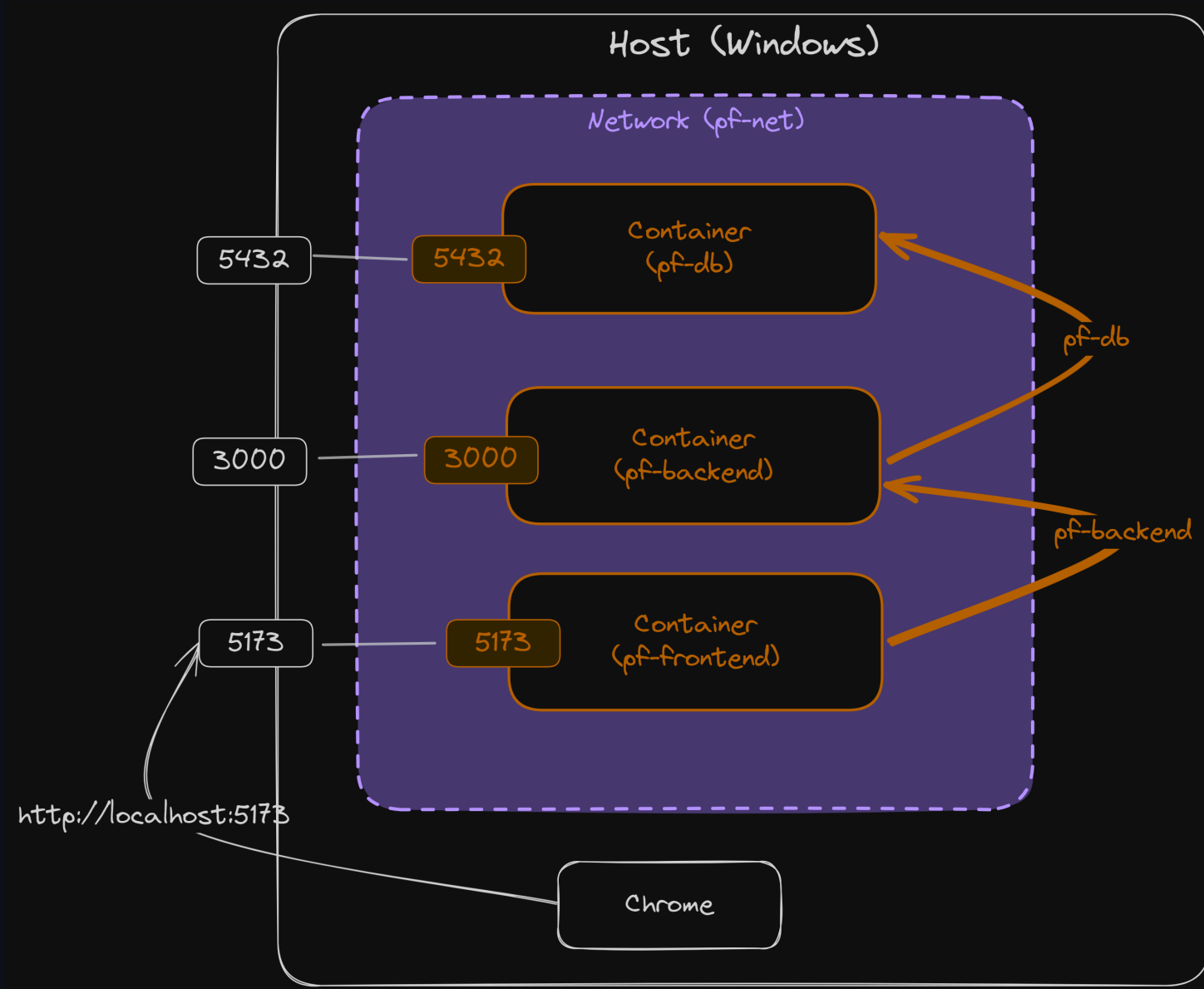


# With reverse proxy





# Containerization and testing

-  Make `./.env.test` from `./.env.test.example` ([Link](#))
-  `./Dockerfile` ([Link](#))
-  `./.dockerignore` ([Link](#))
-  `./nginx.conf.template` ([Link](#)) ([Why](#))
-  `./docker-compose.yml` ([Link](#))
-  `docker compose --env-file ./.env.test up -d --force-recreate --build`



# Push to Dockerhub

- Create an account at <https://hub.docker.com>.
- Create repository called `preflight-frontend`.
- Tag image
  -  `docker tag preflight-frontend [DOCKERHUB_ACCOUNT]/preflight-frontend:latest`
- Push image
  -  `docker push [DOCKERHUB_ACCOUNT]/preflight-frontend:latest`