

# Fullstack Development

# Preflight project - backend

[Github Repo](#)

# Backend applications

- Data storage
- Business logic
- Authentication / authorization
- APIs for frontend frameworks (*depending on architecture*)
- Connection to other services

# Node JS backend frameworks

- Github stars
- State of JS 2024

# Packages

- `npm init es6`
- Express JS
  - `pnpm i express cors helmet morgan debug`
- Typescript
  - `pnpm i typescript @tsconfig/node-lts @tsconfig/node-ts tsx tsc-alias`
  - `pnpm i -D @types/cors @types/express @types/debug @types/morgan`  
`@types/node cross-env nodemon`
- ORM
  - `pnpm i drizzle-orm postgres dotenv`
  - `pnpm i -D drizzle-kit`

# Package explanation

Package	Details
<code>express</code>	Express ( <i>duh!</i> )
<code>helmet</code>	Set default HTTP response header
<code>cors</code>	Enable Cross-origin resource sharing (CORS)
<code>morgan</code>	HTTP request logger
<code>debug</code>	Debugging utility
<code>nodemon</code>	Script monitoring during dev

# ORM code

- All [files](#) in `./db` folder.
  - No need to copy `migration` folder.
  - You can change schema.
- `.\drizzle.config.ts` ([Link](#))

# Files

- `./.env` (Copy from [here](#))
- `./nodemon.json` ([Link](#))
- `./.gitignore` ([Link](#))
- `./tsconfig.json` ([Link](#))
- `./.npmrc` (Copy from [here](#))
- Scripts in `package.json` ([Link](#))



# Minimal example

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

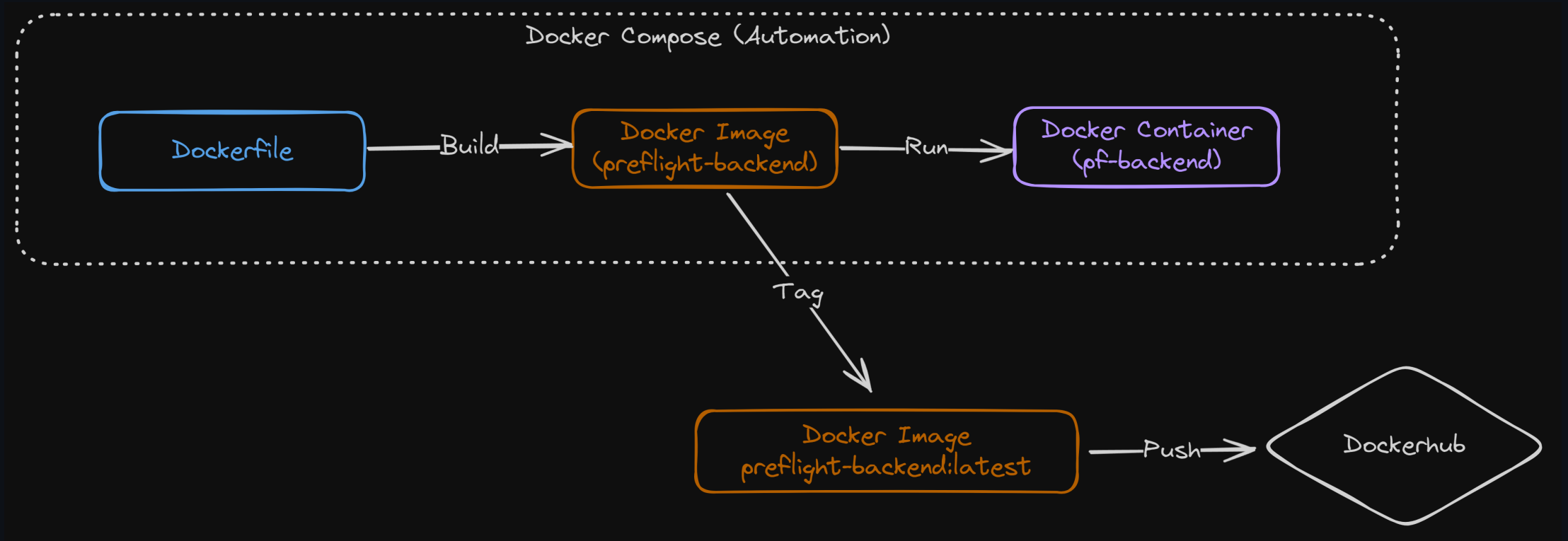
# Full example

- `./src/index.ts` [\(Link\)](#)
- Build
  - `npm run build`
- Start production
  - `npm run start`






# Migration

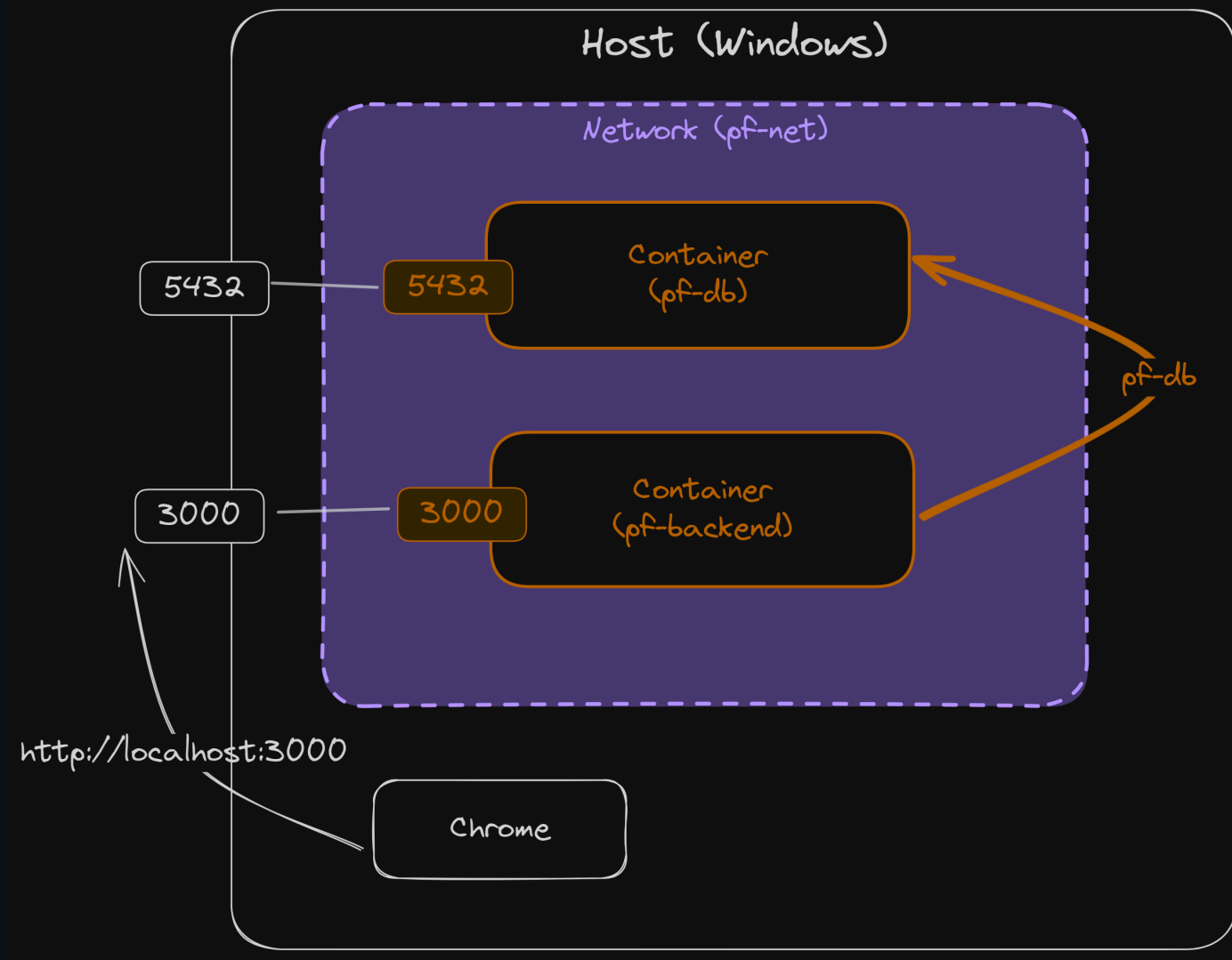
- Run before the containerization step.
- `npm run db:generate`

# Containerization



# Steps

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



# Automated database migration

- We want to make sure that database migrations are run automatically every time the container starts.
- We use the `post_start` hook in `docker-compose.yml`




# Dockerhub

- Create an account at <https://hub.docker.com>.
- Create repository called `preflight-backend`.

# Push to Dockerhub

- Tag image

-  `docker tag preflight-backend [DOCKERHUB_ACCOUNT]/preflight-backend:latest`

- `docker login`

- Push image

-  `docker push [DOCKERHUB_ACCOUNT]/preflight-backend:latest`

# Useful docker commands

- Inspect

- `docker ps`
- `docker network ls`
- `docker volume ls`

- Cleaning

- `docker image prune -a`
- `docker builder prune`
- `docker volume prune`
- `docker network prune`