02. Installation

Sure provides a great installation page on their website as can be found over here. However, I would like to drop the docker compose file straight away. Also note that I like to keep all the volumes inside the same folder instead of all in docker volumes. Hence, this has been changed compared to the original docker compose file. We will deploy this again nicely in Dockhand with compose and .env file that can be found below

compose.yml

```
x-db-env: &db_env
  POSTGRES_USER: ${POSTGRES_USER:-sure_user}
  POSTGRES_PASSWORD: ${POSTGRES_PASSWORD:-sure_password}
  POSTGRES_DB: ${POSTGRES_DB:-sure_production}

x-rails-env: &rails_env
  <<: *db_env
  SECRET_KEY_BASE: ${SECRET_KEY_BASE:-
a7523c3d0ae56415046ad8abae168d71074a79534a7062258f8d1d51ac2f76d3c3bc86d86b6b0b307df30d9a6a
90a2066a3fa9e67c5e6f374dbd7dd4e0778e13}
  SELF_HOSTED: "true"
  RAILS_FORCE_SSL: "false"
  RAILS_ASSUME_SSL: "false"
  DB_HOST: db
  DB_PORT: 5432
  REDIS_URL: redis://redis:6379/1
  # NOTE: enabling OpenAI will incur costs when you use AI-related features in the app
(chat, rules).  Make sure you have set appropriate spend limits on your account before
adding this.
#  OPENAI_ACCESS_TOKEN: ${OPENAI_ACCESS_TOKEN}

services:
  web:
    image: ghcr.io/we-promise/sure:stable
    volumes:
      - ./app-storage:/rails/storage
```

```
    ports:
      - ${PORT:-3000}:3000
      # To also publish on IPv6 (dual-stack), uncomment the line below AND
      # set BINDING=:: in the environment block. See docs/hosting/docker.md
      # "Binding to IPv6" for details.
      # - "[::]:${PORT:-3000}:3000"
    restart: unless-stopped
    environment:
      <<: *rails_env
      # BINDING: "::"  # Uncomment for IPv6 dual-stack inside the container
    depends_on:
      db:
        condition: service_healthy
      redis:
        condition: service_healthy
    dns:
      - 8.8.8.8
      - 1.1.1.1
    networks:
      - sure_net

  worker:
    image: ghcr.io/we-promise/sure:stable
    command: bundle exec sidekiq
    volumes:
      - ./app-storage:/rails/storage
    restart: unless-stopped
    depends_on:
      db:
        condition: service_healthy
      redis:
        condition: service_healthy
    dns:
      - 8.8.8.8
      - 1.1.1.1
    environment:
      <<: *rails_env
    networks:
      - sure_net
```

```
  db:
    image: postgres:16
    restart: unless-stopped
    volumes:
      - ./postgres-data:/var/lib/postgresql/data
    environment:
      <<: *db_env
    healthcheck:
      test: [ "CMD-SHELL", "pg_isready -U $$POSTGRES_USER -d $$POSTGRES_DB" ]
      interval: 5s
      timeout: 5s
      retries: 5
    networks:
      - sure_net

  backup:
    profiles:
      - backup
    image: prodrigestivill/postgres-backup-local
    restart: unless-stopped
    volumes:
      - ./backups:/backups # Change this path to your desired backup location on the host
machine
    environment:
      - POSTGRES_HOST=db
      - POSTGRES_DB=${POSTGRES_DB:-sure_production}
      - POSTGRES_USER=${POSTGRES_USER:-sure_user}
      - POSTGRES_PASSWORD=${POSTGRES_PASSWORD:-sure_password} # pipelock:ignore
      - SCHEDULE=@daily # Runs once a day at midnight
      - BACKUP_KEEP_DAYS=7 # Keeps the last 7 days of backups
      - BACKUP_KEEP_WEEKS=4 # Keeps 4 weekly backups
      - BACKUP_KEEP_MONTHS=6 # Keeps 6 monthly backups
    depends_on:
      - db
    networks:
      - sure_net

  redis:
```

```
    image: redis:latest
    restart: unless-stopped
    volumes:
      - ./redis-data:/data
    healthcheck:
      test: [ "CMD", "redis-cli", "ping" ]
      interval: 5s
      timeout: 5s
      retries: 5
    networks:
      - sure_net

networks:
  sure_net:
    driver: bridge
```

Then finally you can set your environment variables in the .env file, for the postgres and email details. Just make sure you create again a nice secret key with the command

```
openssl rand -hex 64
```

and then paste it into the .env file for the SECRET_KEY_BASE

.env

```
POSTGRES_USER=usernamedifficult
POSTGRES_PASSWORD=0394jgoipjeroigjoisdjgkljyourcode
POSTGRES_DB=sure_production
SECRET_KEY_BASE=a7523c3d0ae56415046ad8abae168d71074a79534a7062258f8d1d51ac2f76d3c3bc86d86b
6b0b307df30d9a6a90a2066a3fa9e67c5e6f374dbd7dd4e0778e13
PORT=3000
SMTP_ADDRESS=smtp.example.com
SMTP_PORT=587
SMTP_DOMAIN=example.com
SMTP_USERNAME=your-username
SMTP_PASSWORD=your-password
SMTP_AUTHENTICATION=plain
SMTP_ENABLE_STARTTLS_AUTO=true
```

> Note! Make sure that sure has the right writing permissions of 750 for the "sure" folder you are hosting everything in. Otherwise it cannot create a directory for your account.

Most likely you will get an error for because of sure not being able to write to the sure folder. In that case as noted above go into your folder and change the permissions and restart the sure stack.

Then you will be able to reach your sure instance at DOCKERVMIP:Port and create a user account. Now, as with any application you should give it a domain name public or not. If you don't know how to do this check the Reverse Proxy section.

Perfect! Now on the next pages we will discuss some basic settings and things you can do with it.

However, just check the documentation for everything of course.

Revision #4
Created 2026-07-05 23:56:18 UTC by dpasn
Updated 2026-07-12 01:30:26 UTC by dpasn