

2.4 Developer Tools

This book contains all the open source tools that might come hand for developers. That is people that want actually build websites and applications that other people can use again ;)

- [1. Payload CMS](#)
- [2. Automation Tools](#)
 - [1. Ansible](#)
- [3 Forgejo](#)
- [4. Project Management Tools](#)
 - [1. Plane](#)
 - [2. Kan.bn](#)
- [5. The Open Project](#)
 - [1. Installation](#)

1. Payload CMS

Source: [Github](#)

2. Automation Tools

1. Ansible

3 Forgejo

A Self hosted federated alternative for Github

4. Project Management Tools

1. Plane

2. Kan.bn

compose.yml

```
services:
  migrate:
    image: ghcr.io/kanbn/kan-migrate:latest
    container_name: kan-migrate
    networks:
      - kan-network
    environment:
      - POSTGRES_URL=${POSTGRES_URL}
    depends_on:
      postgres:
        condition: service_healthy
    restart: "no"

  web:
    image: ghcr.io/kanbn/kan:latest
    container_name: kan-web
    ports:
      - "${WEB_PORT:-1031}:3000"
    networks:
      - kan-network
    env_file:
      - .env
    environment:
      - NEXT_PUBLIC_BASE_URL=${NEXT_PUBLIC_BASE_URL}
      - BETTER_AUTH_SECRET=${BETTER_AUTH_SECRET}
      - POSTGRES_URL=${POSTGRES_URL}
      - NEXT_PUBLIC_ALLOW_CREDENTIALS=true
    depends_on:
      migrate:
```

```
    condition: service_completed_successfully
restart: unless-stopped
```

```
postgres:
```

```
  image: postgres:15
```

```
  container_name: kan-db
```

```
  environment:
```

- POSTGRES_DB=kan_db
- POSTGRES_USER=kan
- POSTGRES_PASSWORD=\${POSTGRES_PASSWORD}

```
  ports:
```

- 5432:5432

```
  volumes:
```

- ./kan_postgres_data:/var/lib/postgresql/data

```
  healthcheck:
```

```
    test: ["CMD-SHELL", "pg_isready -U kan -d kan_db"]
```

```
    interval: 5s
```

```
    timeout: 5s
```

```
    retries: 10
```

```
  restart: unless-stopped
```

```
  networks:
```

- kan-network

```
networks:
```

```
  kan-network:
```

```
#volumes:
```

```
# kan_postgres_data:
```

.env

```
# https://github.com/kanbn/kan?tab=readme-ov-file#environment-variables-)
```

```
# Required environment variables
```

```
NEXT_PUBLIC_BASE_URL= https://kan.rfeyn.org# e.g. https://kan.bn
BETTER_AUTH_SECRET= # Random 32+ char string (can gen with: openssl rand -base64 26 | tr -
dc 'a-zA-Z0-9' | head -c 32)

# Fill if you want to use an external database
POSTGRES_URL= # e.g.
postgresql://kan:asdjf12983rjioskldjf92dojlsdjf9023ioj@your_host:5432/kan

# Only required if deploying from compose
POSTGRES_PASSWORD=asdjf12983rjioskldjf92dojlsdjf9023ioj

# SMTP (optional)
SMTP_HOST=smtp.porkbun.com
SMTP_PORT=587
SMTP_USER=dp@p2pnode.org
SMTP_PASSWORD=
EMAIL_FROM= # e.g. "Kan dp@p2pnode.org"
SMTP_SECURE=false # set to "false" to use port 587
SMTP_REJECT_UNAUTHORIZED= # set to "false" to accept invalid certs

# Switch email features off entirely (optional)
NEXT_PUBLIC_DISABLE_EMAIL=

# S3 storage (optional)
S3_REGION= # required for AWS S3 (e.g. us-east-1); ignored by most S3-compatible providers
– defaults to us-east-1 if unset
S3_ENDPOINT= # set for S3-compatible providers (MinIO, R2, Spaces, etc.); leave blank for
AWS S3
S3_ACCESS_KEY_ID=
S3_SECRET_ACCESS_KEY=
S3_FORCE_PATH_STYLE=
# S3_AVATAR_UPLOAD_LIMIT=2097152 # Avatar upload size limit in bytes (default: 2MB)
NEXT_PUBLIC_STORAGE_URL=
NEXT_PUBLIC_AVATAR_BUCKET_NAME=
NEXT_PUBLIC_ATTACHMENTS_BUCKET_NAME=
NEXT_PUBLIC_STORAGE_DOMAIN=
NEXT_PUBLIC_USE_VIRTUAL_HOSTED_URLS=
```

```
# Auth config (optional)
NEXT_PUBLIC_ALLOW_CREDENTIALS=
NEXT_PUBLIC_DISABLE_SIGN_UP=

# API configuration (optional)
NEXT_API_BODY_SIZE_LIMIT= # e.g. 50mb (defaults to 1mb)

# Integration providers (optional)
TRELLO_APP_API_KEY=
TRELLO_APP_SECRET=

# Redis (optional - for rate limiting)
# If not provided, rate limiting will use in-memory storage
REDIS_URL= # e.g. redis://default:your_password@your_host:6379

# Logging (optional - debug, info, warn, error; defaults to debug in dev, info in prod)
LOG_LEVEL=

# OAuth providers (optional)
BETTER_AUTH_TRUSTED_ORIGINS=
# Optional: Restrict OIDC/Social sign-ins to specific email domains (comma-separated)
BETTER_AUTH_ALLOWED_DOMAINS=
GOOGLE_CLIENT_ID=
GOOGLE_CLIENT_SECRET=
DISCORD_CLIENT_ID=
DISCORD_CLIENT_SECRET=
GITHUB_CLIENT_ID=
GITHUB_CLIENT_SECRET=
GITLAB_CLIENT_ID=
GITLAB_CLIENT_SECRET=
GITLAB_ISSUER=
MICROSOFT_CLIENT_ID=
MICROSOFT_CLIENT_SECRET=
TWITTER_CLIENT_ID=
TWITTER_CLIENT_SECRET=
KICK_CLIENT_ID=
KICK_CLIENT_SECRET=
ZOOM_CLIENT_ID=
ZOOM_CLIENT_SECRET=
```

DROPBOX_CLIENT_ID=
DROPBOX_CLIENT_SECRET=
VK_CLIENT_ID=
VK_CLIENT_SECRET=
LINKEDIN_CLIENT_ID=
LINKEDIN_CLIENT_SECRET=
REDDIT_CLIENT_ID=
REDDIT_CLIENT_SECRET=
ROBLOX_CLIENT_ID=
ROBLOX_CLIENT_SECRET=
SPOTIFY_CLIENT_ID=
SPOTIFY_CLIENT_SECRET=
TIKTOK_CLIENT_ID=
TIKTOK_CLIENT_SECRET=
TIKTOK_CLIENT_KEY=
TWITCH_CLIENT_ID=
TWITCH_CLIENT_SECRET=
APPLE_CLIENT_ID=
APPLE_CLIENT_SECRET=
APPLE_APP_BUNDLE_IDENTIFIER=

5. The Open Project

- **E2EE = NO**
- **Authentication:** Password, 2FA
- **Admin Metadata Access:**
- **Website:** openproject
- **Repository:** [Github](#)
- **Installation Documentation:** [Docker Deployment](#)
- **Installation Videos:** [Opensource is Awesome](#)
- **Community:**
- **Donations:**

1. Installation

Installation Documentation: [Website](#)

According to their official documentation page we can install the Open Project as follows

1. Open up your Docker VM you want to install the open project on
2. Go inside your Arcane projects folder e.g /home/user/arcane/arcane-data/projects
3. Then clone the whole open project repository in there with the following command (Have Git installed)

```
git clone https://github.com/opf/openproject-docker-compose.git --depth=1 --branch=stable/17
openproject
```

4.