🛠️ SimpleFileServer 软件配置与运维文档

1. 配置管理

1.1 环境配置文件

- 所有环境变量应统一配置在.env文件中

示例内容：

```
BASE_DIRECTORY='F:/simple'
```

1.2 配置加载模块

- 使用 `config.js` 加载当前环境的配置项。
- 部分代码片段：

```
// Configuration for the file server
// You can customize the base directory to serve files from
require('dotenv').config();
const fs = require('fs');
const os = require('os');
const path = require('path');
const crypto = require('crypto');

const TMP_DIR = path.join(os.tmpdir(), 'simple-file-server');
const BASE_DIR = process.env.BASE_DIRECTORY || 'F:/simple';
const DB_NAME = crypto.createHash('sha256').update(BASE_DIR.endsWith('/') ?
BASE_DIR.slice(0, -1) : BASE_DIR).digest('hex') + '.db';

if (!fs.existsSync(TMP_DIR)) {
  fs.mkdirSync(TMP_DIR, { recursive: true });
}

module.exports = {

  port: process.env.PORT || 11073,

  baseDirectory: BASE_DIR,
  logsDirectory: process.env.LOG_DIRECTORY || 'logs',

  // Background image path - can be absolute or relative to server root
  backgroundImagePath: process.env.BACKGROUND_IMAGE_PATH || path.join(__dirname,
'bg.jpg'),
  // Background images folder - can be absolute or relative to server root
  backgroundImagesDir: process.env.BACKGROUND_IMAGES_DIR || path.join(__dirname,
'backgrounds'),

  uploadCountLimit: process.env.UPLOAD_COUNT_LIMIT || 10,
  uploadSizeLimit: process.env.UPLOAD_SIZE_LIMIT || 1024 * 1024 * 100, // 100MB

  contentMaxSize: process.env.CONTENT_MAX_SIZE || 5 * 1024 * 1024, // 5MB
```

```
  generateThumbnail: process.env.GENERATE_THUMBNAIL === 'true' || false,
  thumbnailCacheDir: process.env.THUMBNAIL_CACHE_DIR || path.join(TMP_DIR,
'thumbnails')
  }
```

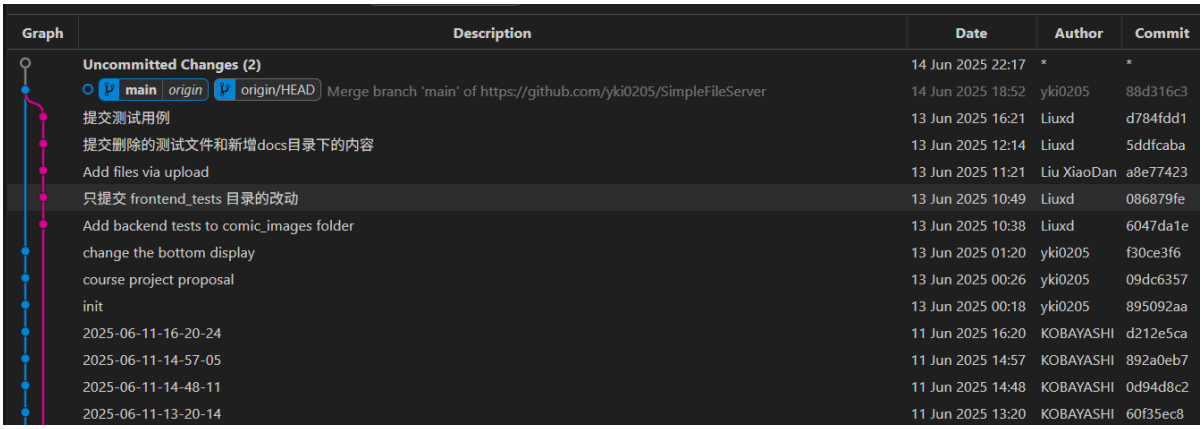
1.3 敏感信息管理

- 敏感信息（如 JWT 密钥）不应硬编码在代码中，应通过环境变量注入。
- CI/CD 构建时使用加密密钥管理工具（如 GitHub Secrets）注入敏感配置。

---

2. 版本控制（Git）

这个软件项目的开发我们采用git丰富的版本控制机制来进行多人协作开发



Graph	Description	Date	Author	Commit
	Uncommitted Changes (2)	14 Jun 2025 22:17	*	*
	main origin origin/HEAD Merge branch 'main' of https://github.com/yki0205/SimpleFileServer	14 Jun 2025 18:52	yki0205	88d316c3
	提交测试用例	13 Jun 2025 16:21	Liuxd	d784fdd1
	提交删除的测试文件和新增docs目录下的内容	13 Jun 2025 12:14	Liuxd	5ddfcaba
	Add files via upload	13 Jun 2025 11:21	Liu XiaoDan	a8e77423
	只提交 frontend_tests 目录的改动	13 Jun 2025 10:49	Liuxd	086879fe
	Add backend tests to comic_images folder	13 Jun 2025 10:38	Liuxd	6047da1e
	change the bottom display	13 Jun 2025 01:20	yki0205	f30ce3f6
	course project proposal	13 Jun 2025 00:26	yki0205	09dc6357
	init	13 Jun 2025 00:18	yki0205	895092aa
	2025-06-11-16-20-24	11 Jun 2025 16:20	KOBAYASHI	d212e5ca
	2025-06-11-14-57-05	11 Jun 2025 14:57	KOBAYASHI	892a0eb7
	2025-06-11-14-48-11	11 Jun 2025 14:48	KOBAYASHI	0d94d8c2
	2025-06-11-13-20-14	11 Jun 2025 13:20	KOBAYASHI	60f35ec8

---

3. 持续集成（CI）

3.1 CI 工具

- GitHub Actions

3.2 CI 流程设计

1. 安装依赖

```
bash

npm ci
```

2. 运行静态检查

```
bash

npm run lint
```

3. 单元测试

```
bash

npm run test
```

4. 构建

```
bash

npm run build
```

5. 打包 Docker 镜像

```
bash

docker build -t simplefileserver .
```

6. 推送镜像

```
bashdocker tag simplefileserver registry.example.com/simplefileserver:latest
docker push registry.example.com/simplefileserver:latest
```

3.3 GitHub Actions 示例工作流

```
yamlname: Build & Deploy

on:
  push:
    branches:
      - main

jobs:
  build:
    runs-on: ubuntu-latest
    steps:
      - name: Checkout code
        uses: actions/checkout@v3

      - name: Setup Node.js
        uses: actions/setup-node@v3
        with:
          node-version: 18

      - name: Install dependencies
        run: npm ci

      - name: Lint
        run: npm run lint

      - name: Test
        run: npm run test
```

```
      - name: Build
        run: npm run build

      - name: Deploy to server
        uses: appleboy/ssh-action@master
        with:
          host: ${{ secrets.HOST }}
          username: ${{ secrets.USERNAME }}
          password: ${{ secrets.PASSWORD }}
          script: |
            cd /var/www/simplefileserver
            git pull origin main
            npm install
            pm2 restart dist/server.js
```

---

4. 部署策略

4.1 本地部署

工具需求

- Node.js 18+ and npm

安装依赖

```
# Install backend dependencies
cd backend
npm install

# Install frontend dependencies
cd ../frontend
npm install
```

配置环境变量

```
BASE_DIRECTORY='F:/simple'
```

前端启动服务

```
bash
(Default port: 2711)
cd frontend
npm run build
npm start
```

后端启动服务

```
# Run the backend server (development mode) (Default port: 11073)
cd backend
npm run dev
```

打开浏览器访问应用

```
http://localhost:${YOUR_PORT_HERE}
```

4.2 Docker 部署

构建镜像

```
bash

docker build -t simplefileserver .
```

启动容器

```
bashdocker run -d \
  -p 3000:3000 \
  -v ./uploads:/app/uploads \
  -e PORT=3000 \
  --name simplefileserver \
  simplefileserver
```

---

5. 运维计划

5.1 日志监控

- 访问控制台打印输出的错误日志

5.2 性能监控

- 集成 Prometheus + Grafana 监控系统性能指标
- 使用 `express-prom-bundle` 收集 HTTP 请求数据

5.3 自动更新机制

- 使用 `nodemon` 实现热重载（开发环境）
- ```
  pm2 start dist/server.js -i max --no-daemon
  ```

5.4 备份与恢复

- 定期备份
- 使用脚本或云存储实现定时备份

```
tar -czf backup_$(date +%Y%m%d).tar.gz uploads/
```
```

5.5 安全加固

- 强制 HTTPS（现在使用的还是http）
- 防止目录遍历攻击（验证请求路径）
- 设置上传文件大小限制和类型白名单
- 使用 Helmet 中间件增强安全头设置