

## مستند پروژه «روشن نیوز»

### مقدمه:

پروژه‌ی «روشان نیوز» یک سامانه‌ی جمع‌آوری، ذخیره‌سازی و نمایش اخبار از سایت‌های فارسی است. در این پروژه با استفاده از تکنولوژی‌های مدرن مانند Docker، Celery، Scrapy و Django، یک سیستم کامل ساختیم که اخبار را از سایت Zoomit جمع‌آوری کرده، آن‌ها را پردازش کرده و از طریق یک API در اختیار کاربران قرار می‌دهد.

### ابزارهای استفاده‌شده:

ابزار	کاربرد
Python 3.12	زبان برنامه‌نویسی اصلی
Django	فریم ورک اصلی برای ساخت API و مدیریت داده
Scrapy	فریم‌ورک کراول و استخراج اخبار از وب‌سایت
Playwright	اجرای کراول مبتنی بر مرورگر برای صفحات داینامیک
Celery	جرای تسک‌های غیربلوک‌کننده، مانند ذخیره اخبار
Celery Beat	زمان‌بندی اجرای خودکار تسک‌های Celery
Redis و	Celery برای Message broker
Flower	مانیتورینگ تسک‌های Celery
Docker	کانتینری‌سازی کل پروژه برای توسعه و اجرا در محیط ایزوله
Docker Compose	مدیریت چند سرویس Docker با یک فایل مشترک

### فرایند جمع‌آوری خبر:

1. اجرای دستور `docker compose up —build -d` کانتینر شروع به اجرا میکند.
2. Playwright صفحه‌های جاوااسکریپتی را پردازش می‌کند.
3. داده‌های استخراج‌شده از طریق یک درخواست POST به مسیر `http://web:8000/api/news/insert` فرستاده می‌شوند.
4. در سمت سرور، این درخواست در یک تسک Celery مدیریت شده و خبر در دیتابیس ذخیره می‌شود.
5. تمامی اخبار از طریق api زیر در دسترس هستند:

[“http://127.0.0.1:8000/api/news/”](http://127.0.0.1:8000/api/news/)

همچنین امکان فیلتر کردن اخبار بر اساس 3 شاخص tag, keyword, exclude\_keyword وجود دارد

[“http://127.0.0.1:8000/api/news/?keyword=\[عبارت مورد نظر\]”](http://127.0.0.1:8000/api/news/?keyword=[عبارت مورد نظر]/)

[“http://127.0.0.1:8000/api/news/?tags=\[عبارت مورد نظر\]”](http://127.0.0.1:8000/api/news/?tags=[عبارت مورد نظر]/)

[“http://127.0.0.1:8000/api/news/?exclude\\_keyword=\[عبارت مورد نظر\]”](http://127.0.0.1:8000/api/news/?exclude_keyword=[عبارت مورد نظر]/)

و در نهایت امکان ترکیب کردن فیلتر ها نیز وجود دارد

[http://127.0.0.1:8000/api/news/?exclude\\_keyword=\[عبارت مورد نظر\]&keyword=\[عبارت مورد نظر\]](http://127.0.0.1:8000/api/news/?exclude_keyword=[عبارت مورد نظر]&keyword=[عبارت مورد نظر]/)

6. برای ذخیره سازی اخبار جدید گفته شد از api زیر استفاده میکنیم

[“http://127.0.0.1:8000/api/news/insert/”](http://127.0.0.1:8000/api/news/insert/)

## تسک‌های Celery:

فایل tasks.py شامل تسک مربوط به کرال و ذخیره سازی اخبار است. این تسک‌ها توسط Scrapy در بازه زمانی ثابت توسط Beat اجرا می‌شوند تا فشار پردازش از مسیر اصلی Django برداشته شود.

## زمان‌بندی (Celery Beat):

با Celery Beat، می‌توانیم جمع‌آوری اخبار را هر ساعت یا هر زمان خاصی به صورت خودکار انجام دهیم. تنظیمات مربوطه در docker-compose.yml برای Beat لحاظ شده‌اند.

## Flower – مانیتورینگ Celery:

در Docker Compose یک سرویس flower هم تعریف شده تا از طریق مرورگر بتوانیم وضعیت تسک‌ها را مشاهده کنیم:

<http://localhost:5555>

## اجرای پروژه با Docker:

### 1. ساخت و اجرا:

```
docker compose up --build -d
```

2. برای مشاهده لاگ‌های سرویس‌های مختلف نیز میتوان از دستورات زیر استفاده کرد

```
docker compose logs -f web
```

```
docker compose logs -f celery
```

```
docker compose logs -f celery-beat
```

### 3. متوقف‌سازی:

```
docker compose down
```