

فاز سوم :

اضافه کردن بخش proxysql به فایل داکر کامپوز

**Services** : این بخش حاوی تعریف‌های مربوط به سرویس‌های مختلف است. هر سرویس دارای یک نام، نسخه مورد استفاده مانند `wordpress:latest` یا `nginx:latest`، تنظیمات شبکه و دیگر تنظیمات است.

**Nginx** : یکی از سرویس‌های تعریف شده که از تصویر `nginx:latest` استفاده می‌کند. این سرویس برای مدیریت ترافیک وب و ایجاد اتصالات HTTP به سرویس‌های دیگر مورد استفاده قرار می‌گیرد. پورت ۸۰ برای ترافیک HTTP از طریق پورت ۸۰ سرویس `nginx` برای ارتباط تعریف شده است.

**WordPress** : دو نمونه از سرویس وردپرس که از تصویر `wordpress:latest` استفاده می‌کنند. این سرویس‌ها دو نسخه مستقل از سایت وبلاگ یا وبسایت ایجاد می‌کنند و از دو حجم داده `wordpress_data` برای ذخیره داده‌های وردپرس استفاده می‌کنند.

**MariaDB** : دو نمونه از سرویس دیتابیس `MariaDB` (یکی اصلی و دیگری تکرار کننده) که از تصویر `mariadb:latest` استفاده می‌کنند. این سرویس‌ها از دو حجم داده `master-data` و `replica-data` برای ذخیره داده‌های دیتابیس استفاده می‌کنند. برای دیتابیس تکرار کننده `MariaDB`، پارامتر `--replicate-do-db=mydb --server-id=2` برای همگام‌سازی داده‌ها تعیین شده است.

**ProxySQL** : یک سرویس `ProxySQL` برای توزیع بار بین دیتابیس‌ها و بهبود عملکرد سیستم دیتابیس تعریف شده است. این سرویس از نسخه `proxysql` استفاده می‌کند. این سرویس از دو سرویس `MariaDB` به عنوان ورودی به `ProxySQL` به عنوان وابستگی‌ها استفاده می‌کند.

**Volumes** : در این بخش، حجم‌های داده برای ذخیره اطلاعات مورد نیاز هر سرویس تعریف شده‌اند. این اطلاعات در مسیرهای `/var/www/html` برای وردپرس و `/var/lib/mysql` برای دیتابیس‌ها ذخیره می‌شوند.

**networks** : تنظیمات شبکه مورد استفاده برای ارتباط بین سرویس‌ها و ایجاد یک شبکه سفارشی تعریف شده است. این شبکه از نوع `bridge` است و آدرس‌دهی آن بر روی زیرشبکه `۲۸/۱۷۲.۲۸.۰.۰` صورت می‌گیرد

```
4 services:
53 # ProxySQL service for database load balancing
54 proxysql:
55   image: proxysql/proxysql
56   volumes:
57     - ./proxysql.cnf:/etc/proxysql.cnf # Mount ProxySQL configuration
58   ports:
59     - "6033:6033" # Expose port 6033 for ProxySQL
60   depends_on:
61     - mariadb-master # Depend on the master MariaDB instance
62     - mariadb-replica # Depend on the replica MariaDB instance
63   networks:
64     - cangrownet # Use the cangrownet network
65   env_file: env, # Load environment variables from the 'env' file
66
67 # Define volumes for persistent data storage
68 volumes:
69   wordpress_data: # Volume for WordPress data
70   master-data: # Volume for master MariaDB data
71   replica-data: # Volume for replica MariaDB data
72
73 # Define network configuration
74 networks:
75   cangrownet: # Define a custom bridge network
```

فاز چهارم :

ابتدا برای انجام فاز چهارم یکی از تم های وردپرس را انتخاب کردم و اون تم رو دانلود کردم و فایل هاش رو از طریق سیستم به گیت هاب منتقل کردم سپس یک فایل به اسم `main.yml` در ریپازیتوری پروژه در پوشه ی `github/workflows` ایجاد کردم :

`name: CI/CD Pipeline`

`on:`

`push:`

`branches:`

`- main`

`jobs:`

`deploy:`

`runs-on: ubuntu-latest`

`steps:`

`- name: Checkout repository`

`uses: actions/checkout@v2`

`- name: Deploy theme`

`run: |`

`cp -r /github/Cangrow-Project/twentytwentytwo http://185.240.151.74/wp-admin/themes.php?theme=twentytwentytwo`

این فایب برای یک فرآیند CI/CD است. در این فرآیند، هر زمان که تغییری در شاخه `main` روی مخزن صورت می گیرد، یک وظیفه با نام `deploy` اجرا می شود.

وظیفه `deploy` در یک محیط اجرایی `Ubuntu` آخرین نسخه اجرا می شود. این وظیفه دارای دو گام است:

`Checkout repository` : این گام از اکشن `actions/checkout@v2` برای کپی کردن محتویات مخزن فعلی استفاده می کند.

`Deploy theme` : این گام یک فرآیند است که تم را از مسیر `github/Cangrow-Project/twentytwentytwo/` به مسیر `http://185.240.151.74/wp-admin/themes.php?theme=twentytwentytwo` کپی می کند. این فرآیند برای بروزرسانی تم در یک وبسایت `WordPress` استفاده می شود. این فرآیند به صورت `shell script` با استفاده از دستور `cp` اجرا می شود.

چالش فاز چهارم :

من این فایل برای اجرای خودکار کدها و اعمال تغییرات آن به هاست نوشتم و کل این مراحل تا اینجا انجام دادم. اما وقتی این وظایف را اجرا می‌کنم، با ارور زیر مواجه می‌شوم که می‌گوید نمی‌تواند فایل‌هایی که تغییرات در آنها ایجاد شده است را به هاست منتقل کند زیرا مسیر آنها پیدا نمی‌شود.

The screenshot shows the GitHub Actions interface for a workflow named 'main.yml' (run #6). The workflow is currently in a failed state, indicated by a red 'x' icon. The left sidebar shows the 'Summary' tab selected, with a list of jobs including 'deploy'. The main panel displays the 'deploy' job details, which failed 51 minutes ago. The job steps are listed as follows:

- Set up job (1s)
- Checkout repository (1s)
- Deploy theme (0s) - This step failed.
- Post Checkout repository (0s)
- Complete job (0s)

The error message for the 'Deploy theme' step is:

```
1 ▶ Run cp -r /github/Cangrow-Project/twentytwentytwo http://185.249.151.74/wp-admin/themes.php?theme=twentytwentytwo
4 cp: cannot stat '/github/Cangrow-Project/twentytwentytwo': No such file or directory
5 Error: Process completed with exit code 1.
```