



ΤΕΧΝΟΛΟΓΙΕΣ ΝΕΦΟΥΣ

ΑΠΑΛΛΑΚΤΙΚΗ ΕΡΓΑΣΙΑ ΕΡΓΑΣΤΗΡΙΟΥ - ΑΤΟΜΙΚΗ

Τριανταφύλλης Πέτρος Ευάγγελος - 321/2017193

Καθηγητής : Θ. Κρητικός

15/6/2023

Περιεχόμενα

Λεπτομέρειες υλοποίησης	2
Τεχνολογικό στάκ:	2
Οδηγίες Εγκατάστασης.....	3
Παράμετροι εφαρμογής	3
Παραδοτέα	5
Ζήτημα Πρώτο (20/100)	5
Ζήτημα Δεύτερο (25/100).....	5
Ζήτημα Τρίτο (25/100)	5
.....	5
Ζήτημα Τέταρτο (25/100).....	5
Εμπειρία που αποκομίστηκε.....	6

Λεπτομέρειες υλοποίησης

Τεχνολογικό στάκ:

Εργαλεία και Περιβάλλοντα:

- IntelliJ IDEA (ενσωματωμένο περιβάλλον ανάπτυξης για τη γλώσσα Java)
- Docker Compose (εργαλείο για τη διαχείριση πολλαπλών Docker containers)
- Kubernetes (ανοιχτού κώδικα σύστημα διαχείρισης εφαρμογών που εκτελούνται σε containers)

Backend:

- Java (γλώσσα προγραμματισμού)
- Spring Boot (πλαίσιο ανάπτυξης εφαρμογών)
- Hibernate (αντικειμενοστραφής αντιστοίχιση αντικειμένου-σχέσης για την πρόσβαση στη βάση δεδομένων)

Δοχεία (Containers):

- Docker (για την δοχειοποίηση και απομόνωση των εφαρμογών και της βάσης δεδομένων)

Σύστημα Διαχείρισης Βάσης Δεδομένων:

- MySQL (σύστημα διαχείρισης βάσης δεδομένων) | user:root , password : pass1312

Οδηγίες Εγκατάστασης

Προαπαιτούμενα:

1. Βεβαιωθείτε ότι έχετε εγκαταστήσει τα εξής στον υπολογιστή σας:
 - Docker: Ακολουθήστε τις επίσημες οδηγίες για την εγκατάσταση του Docker στο συγκεκριμένο λειτουργικό σύστημα που χρησιμοποιείτε.
 - Docker Compose: Ακολουθήστε τις επίσημες οδηγίες για την εγκατάσταση του Docker Compose στο συγκεκριμένο λειτουργικό σύστημα που χρησιμοποιείτε.

Οδηγίες εγκατάστασης και εκτέλεσης:

2. Λήψη του κώδικα:
 - Κατεβάστε τον κώδικα του προτζέκτ από το αποθετήριο του στο GitHub ή τον τοπικό φάκελο όπου έχετε αποθηκεύσει τον κώδικα.
3. Μεταβείτε στον φάκελο όπου βρίσκονται τα αρχεία docker-compose.yml, start.sh, stop.sh και setup.sh.
4. Εκτελέστε το αρχείο setup.sh
5. Αφού ολοκληρωθεί η εγκατάσταση, μπορείτε να εκτελέσετε το σύστημα εκτελώντας το start.sh. Αυτό το script θα εκτελέσει τους Docker containers και θα ξεκινήσει την εφαρμογή.
6. Προσπελάστε την εφαρμογή: Αφού η εφαρμογή έχει ξεκινήσει, μπορείτε να αποκτήσετε πρόσβαση σε αυτή μέσω <http://localhost:8080> και να χτυπήσετε τα ανάλογα endpoints.
7. Για να σταματήσετε την εφαρμογή, εκτελέστε το αρχείο stop.sh:

Παράμετροι εφαρμογής

src/main/resources/application.properties

```
spring.datasource.url=jdbc:mysql://172.17.0.1:3306/dvds
spring.datasource.username=root
spring.datasource.password=pass1312

spring.datasource.driver-class-name=com.mysql.cj.jdbc.Driver

# Hibernate Configuration
spring.jpa.hibernate.ddl-auto=update
spring.jpa.properties.hibernate.dialect=org.hibernate.dialect.MySQLDialect
|
```

Σε αυτό το αρχείο έχουμε παραμετροποιήσει ότι αφορά τη σύνδεση της spring εφαρμογής με τη βάση και χρήση Hibernate. Η ip της βάσης ανιχνευθηκε με χρήση της εντολής docker network inspect.

Παραδοτέα

Ζήτημα Πρώτο (20/100)

1. Παραδοτέος πηγαίος κώδικας spring boot project : `src/main/java`
2. Παραδοτέο αρχείο migration `.sql` : `migrationv1.sql`
3. Παραδοτέο αρχείο `.properties` : `src/main/resources/application.properties`

Ζήτημα Δεύτερο (25/100)

1. Αρχείο `Dockerfile` : `Dockerfile` , βρίσκεται στο root του project.
2. Αρχεία scripts : `setup.sh` - `start.sh` - `stop.sh`, βρίσκονται στο root του project.

Ζήτημα Τρίτο (25/100)

Το αρχείο `docker-compose.yml` βρίσκεται στο root του project

Ζήτημα Τέταρτο (25/100)

Για αυτό το ερώτημα έχουν συνταχθεί τα παρακάτω αρχεία:

- `yaml/db-persistent-volume.yaml`
- `yaml/dvddatabase-deployment.yaml`
- `yaml/dvddatabase-service.yaml`
- `yaml/dvdservice-deployment.yaml`
- `yaml/dvdservice-service.yaml`

Εμπειρία που αποκομίστηκε.

Κατά τη διάρκεια του project, αντιμετωπίσαμε ορισμένα προβλήματα συνδεσμολογίας με τη βάση δεδομένων. Οι τεχνικές λεπτομέρειες και οι εντολές που χρησιμοποιήθηκαν για την αντιμετώπιση αυτών των προβλημάτων ποικίλλουν ανάλογα με την αιτία του προβλήματος. Γενικά κάποιες εντολές που χρησιμοποιούσαμε συχνά ήταν :

- **docker ps (-a όλα):** για να ελέγξουμε την κατάσταση των εκτελούμενων containers.
- Αν ένα container δεν εκτελείται χρησιμοποιούμε το **docker logs [container_id]** για να δούμε το σφάλμα που επιστρέφει η εφαρμογή.
- **docker inspect [container_id]:** για να λάβουμε πληροφορίες σχετικά με το εκάστοτε container. Ελέγχουμε παραμέτρους του δικτύου του container, όπως τη διεύθυνση IP και τη θύρα που χρησιμοποιείται για τη σύνδεση με τη βάση δεδομένων.
- Επανεκκίνηση των Docker services με τα ήδη υπάρχοντα script που έχουμε συντάξει.

Όσον αφορά τη συνολική εμπειρία από το project, κάποιες κύριες πτυχές περιλαμβάνουν:

- Κατανόηση της δομής και της λειτουργίας του συστήματος: Κατά την ανάπτυξη του project, αναλύσαμε τις απαιτήσεις και σχεδιάσαμε τη δομή του συστήματος, συμπεριλαμβανομένων των Docker containers, των υπηρεσιών και της βάσης δεδομένων.
- Διαχείριση Docker: Κατά την εγκατάσταση και την εκτέλεση των Docker containers, αποκτήσαμε εμπειρία στη διαχείριση του Docker και των συναφών εργαλείων. Μάθαμε πώς να δημιουργούμε, να εκτελούμε και να συντηρούμε τα containers.
- Ανάπτυξη εφαρμογών: Αναπτύξαμε το σύστημα χρησιμοποιώντας διάφορες τεχνολογίες, όπως Java, Spring Boot, Hibernate, MySQL.
- Αντιμετώπιση προβλημάτων: Κατά τη διάρκεια του project, αντιμετωπίσαμε προβλήματα, όπως προβλήματα συνδεσιμότητας με τη βάση δεδομένων. Από αυτή την εμπειρία, μάθαμε πώς να εντοπίζουμε και να αντιμετωπίζουμε προβλήματα στο περιβάλλον των Docker containers και των υπηρεσιών.

Τριανταφύλλης Πέτρος Ευάγγελος - 321/2017193
Πανεπιστήμιο Αιγαίου.

