Deployment Documentation

Marta Espejo, Pedro Figueredo, Jordi Nadeu

November 29, 2024

Contents

1 Architecture Overview

This document describes the deployment of a multi-service application architecture using Docker Compose. The services include:

- **MariaDB**: Relational database server for Moodle.
- **Moodle**: Learning Management System (LMS).
- **Core App**: A custom-built application.
- **MongoDB**: NoSQL database for the core app.
- **Redis**: In-memory cache to improve application performance.
- **NFS Server**: Shared file storage.
- **Nginx**: Reverse proxy for routing traffic and serving static files.

The deployment ensures scalability, modularity, and efficient data management with persistent storage and a unified internal network.

2 Networking

All services are connected to a shared **custom internal network** called `internal-network`. This ensures secure communication between containers without exposing internal services to the host machine or the public network.

3 Data Management and Backups

3.1 Data Management

- **MariaDB**: Stores relational data in the `mariadb_data` volume.
- **Moodle**: Application data and file uploads are stored in `moodle_data` and `moodledata_data` volumes.
- **MongoDB**: NoSQL data is initialized via a setup script in the `mongo-setup` container.
- **NFS Server**: Uses the `./nfs/nfsdisk` directory for shared files.
- **Static Files**: Nginx serves static content from the `./static/data` directory.

3.2 Backups

- **MariaDB**: Backup using `mysqldump`:

```
docker exec mariadb mysqldump -u bn_moodle -p bitnami_moodle > mariadb
```

- **Moodle Data**: Archive Moodle data directories:

```
tar -czvf moodle_backup.tar.gz ./moodle_data ./moodledata_data
```

- **MongoDB**: Use `mongodump` to create backups:
  ```
  docker exec mongodb mongodump --uri="mongodb://app:secret@localhost:270
  ```
- **NFS**: Backup shared files with `rsync`:
  ```
  rsync -av ./nfs/nfsdisk /path/to/backup/location
  ```

4 Steps to Deploy

Start the services using Docker Compose:

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

4.1 Access Services

- **Nginx**: `http://localhost:8080`
- **Moodle**: `http://localhost:8081`
- **Core App**: Routed via Nginx on `/`.