

Разработать приложение со следующими функциональными возможностями (*могут быть другие, но количество функциональных возможностей должно быть не меньше*):

1. окно авторизации,
2. главная страница: должна содержать информацию о приложении и ссылки на авторизацию или регистрацию,
3. личная страница пользователя: после успешного входа пользователи должны перенаправляться на свою личную страницу, где они смогут видеть информацию о своем профиле,
4. наличие базы данных: приложение должно использовать базу данных для хранения логинов и паролей пользователей. *Опционально: Ведение в базе данных списка пользователей с их состояниями (например, активный, неактивный).*

На оценку «удовлетворительно»

1. Docker-контейнеризация:

- Использовать Docker и docker-compose для развертывания приложения и базы данных
- Создать Dockerfile для контейнеризации приложения, а также docker-compose.yml, который определяет сервисы

На оценку «хорошо»

Развернуть приложение в Kubernetes:

1. Kubernetes Deployment:

- создать манифесты Kubernetes для развертывания приложения и базы данных,
- использовать минимум 3 Pod.

2. Сервисы (*могут быть другие, но количество должно быть не меньше*):

- Сервис аутентификации – отвечает за обработку входа пользователей, регистрацию новых аккаунтов и т.д.
- Сервис профиля – управляет данными профиля пользователя (например, отвечает за получение и обновление информации о пользователе, такой как имя, адрес электронной почты и другая информация)
- Сервис отчетов – агрегирует и формирует отчеты на основе данных пользователей (например, может отвечать за генерирование отчетов о активности пользователей, включая вызов CronJob, который будет выполнять регулярные задания по сбору информации (например, о последних зарегистрированных пользователях)).
- Сервис уведомлений – отвечает за отправку уведомлений пользователям.

3. Использовать Liveness и Readiness Probes для обеспечения устойчивости приложения.

4. Secret для хранения конфиденциальных данных (например, учетные данные для доступа к БД).

5. ConfigMap для изменения заголовка окна авторизации (например, изменить текст "Авторизация" на "Вход в приложение").

На оценку «отлично»

1. CronJob (могут быть другие, но количество должно быть не меньше):

- CronJob для сбора последних 5 зарегистрированных пользователей

CronJob для отправки уведомлений (например, отправляет уведомления пользователям о новых функциях или изменениях в системе каждый день).
- CronJob для очистки устаревших сессий (например, каждый день очищает устаревшие пользовательские сессии)

2*. Использовать (хотя бы попытаться использовать) что-то одно из перечисленного

- добавить в helm все прошлое, сделать маршрутизацию (например, если верные данные, то вход в личную страницу пользователя, иначе на главную),
- добавить какое-нибудь асинхронное взаимодействие (например, Kafka),
- настроить мониторинг сервисов (например, Prometheus, Grafana).