



Итерация 3

0310 Кузнецов Андрей



План текущей итерации

- Чтение данных из БД;
- Реализация интеграционных тестов для репозитория;
- Настройка конфигурации БД из CLI приложения;
- Тестирование абстракций работы с кэшем;
- Интегрирование CLI в приложение;
- Интеграция сервисов извлечения кэша с репозиторием.



Результаты итерации

- Реализована возможность чтения данных из БД;
- Реализованы интеграционные тесты для репозитория;
- Настроена конфигурация БД из CLI приложения;
- Протестированы сервисы работы с кэшем.



Интеграционные тесты для репозитория

Проблема при работе с TestContainers:

- интеграционные тесты не могут запуститься в контейнере без Docker, поскольку TestContainers нужен доступ к API Docker.

Было перепробовано несколько решений, сработало:

- Прокидывание сокета: `--volume /var/run/docker.sock:/var/run/docker.sock`.

Результат:

- Успешное выполнение интеграционных тестов в CI.



Успешное выполнение интеграционных тестов

```
1892 test/unit/codeplag/db/test_mongo.py::TestMongoDBInfrastructure::test_mongodb_connection PASSED [ 19%]  
1893 test/unit/codeplag/db/test_mongo.py::TestReportRepository::test_report_repository_write_and_get PASSED [ 19%]  
1894 test/unit/codeplag/db/test_mongo.py::TestReportRepository::test_report_repository_nonexistent_comparison PASSED [ 20%]  
1895 test/unit/codeplag/db/test_mongo.py::TestFeaturesRepository::test_features_repository_write_and_get PASSED [ 21%]  
1896 test/unit/codeplag/db/test_mongo.py::TestFeaturesRepository::test_features_repository_nonexistent_file PASSED [ 21%]
```



Реализация базового механизма протухания

- При извлечении данных из Mongo сервисы (MongoReporter и MongoFeaturesCache) сверяют sha256 и modify_date соответственно.
- Если поля не совпадают - сервисы возвращают None (данные устарели), начинается перевычисление и уже обновленные данные снова попадают в кэш.
- Например, если изменить файл, не меняя его текстового содержания (например, добавить пустую строку), то запустится перевычисление лишь ASTFeatures, а FullCompareInfo будет взят из кэша (sha256 останется прежним)



Настроенный CLI





- В поле `reports_extension` добавлен вариант `mongo`, который активирует кэширование через Mongo;
- Добавлены варианты для конфигурации хоста, порта, пользователя и пароля БД;
- Дополнены юнит- и автотесты, тестирующие поведение CLI, также проверено отклонение невалидных значений (например, неправильный `reports_extension`, неправильный номер порта)



Пример успешного pipeline

All checks have passed ×

4 successful checks

✓		Check source code and push created image based on sources / check-code (push) Successful in 50s	Details
✓		Collect Commits / collect-commits (push) Successful in 6s	Details
✓		Check source code and push created image based on sources / docker-build-test-autotest (push) Successful in 8m	Details
✓		Check source code and push created image based on sources / push-image (push) Successful in 1m	Details



План на следующую итерацию

- Настройка docker-compose для запуска всего приложения разом;
- Добавление автотестов на проверку записи/чтения из БД через testcontainers;
- Интегрироваться с главной веткой проекта;
- Предусмотреть механизм временного “протухания” кэша.



Скринкаст

- Для ускорения демонстрации работы записи/чтения/протухания кэша используются логи приложения, но похожего результата можно достичь, прочитав логи непосредственно Mongo;
- Чтение из кэша происходит всегда, даже если его данные протухли, поскольку проверка происходит на уровне выше - в таком случае после чтения последует перезапись.

Ссылка на скринкаст: <https://disk.yandex.ru/i/UrMjqvphLvg-sA>