Министерство науки и высшего образования Российской Федерации

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»

(Университет ИТМО)

Факультет: Информационных технологий и программирования

Направление подготовки: 01.03.02 Прикладная математика и информатика

Образовательная программа Информатика и программирование

О Т Ч Е Т

о производственной практике

Тема задания: создание приложения “Smart Reminder” для уведомления в Slack

Обучающейся Карасевой Екатерины Павловны M33351

Руководитель практики от профильной организации: *Кочелаев Дмитрий Юрьевич, Директор департамента разработки и исследований, ООО "Солантек"*

Руководитель практики от университета: *Корнеев Георгий Александрович, доцент ФИТиП*

Санкт-Петербург

2022 г

1. **Цели и задачи практики**

Цели — совершенствование навыков проектирования и разработки программного обеспечения. Приобретение навыков планирования производственных процессов.

Задачи — изучение методов организации производственного процесса. Закрепление знаний, полученных в процессе обучения, посредством применения их в разработке программного обеспечения.

1. **Сведения об организациях**

**Solanteq** — международная компания имеющая филиалы в нескольких городах и странах мира: офисы находятся в Санкт-Петербурге, Москве, Самаре, Уфе, Молдавии, Кипре, Бельгии, и даже Дубаи.

Компания разрабатывает платформу Solar — систему инновационных компонент для построения эффективных и элегантных платежных решений.

Активно развиваясь с момента основания в 2014 году, SOLANTEQ предлагает на мировом рынке мощные, производительные, построенные на современных технологиях программные продукты для развития розничного бизнеса банков и процессинговых центров.

В отличие от большинства других платформ, SOLAR был создан экспертами с учетом самых актуальных вызовов, стоящих перед глобальным миром платежей, таких как продолжающаяся диджитализация, ориентация на клиентский опыт, контроль рисков и оптимизация стоимости.

Среди клиентов (35+) несколько банков первой десятки в РФ, а также банки и процессинговые центры за рубежом.

**Университет ИТМО** (Санкт-Петербург) ― национальный исследовательский университет, ведущий вуз России в области информационных и фотонных технологий. Один из лидеров глобального и национального академического сообщества.

ИТМО — это более 12 500 студентов из порядка 90 стран и свыше 4000 ученых, преподавателей и сотрудников.

В структуре университета ― 4 мегафакультета и 15 факультетов. Приоритетные направления: IT, фотоника, робототехника, квантовые коммуникации, информационная безопасность, трансляционная медицина, умные материалы, химия, биотехнологии, урбанистика.

1. **Занимаемая должность**

Практикант

1. **Цели проекта**

Разработка приложения “Smart Reminder”, реализующего основной функционал:

- Вопросы в чатах поддержки Slack, оставшиеся без ответа. Вопрос считается отвеченным, если по нему есть тред, в котором есть ответы от команды продукта (для каждого чата поддержки перечень пользователей команды продукта задается в конфигурации приложения).

Позднее в отчете по учебной практике в данному приложению добавится дополнительная функциональность:

- Слежение за новыми задачами в Youtrack, отправка онлайн уведомлений о новых задачах пользователям в Slack.

- Дайджест уведомления, включающие в себя:

- Перечень задач с нарушенным SLA из Youtrack.

- Упоминания в задачах Youtrack, оставшиеся без ответа. Упоминание считается отвеченным, если пользователь из команды проекта выполнил одно из действий: ответил в комментариях, поставил реакцию на комментарий, изменил задачу.

1. **Использованные технологии**

- Система контроля версий Git

- Язык программирования Kotlin

- Среда разработки Intellij IDEA

- Slack API -- Slack Bolt for JVM

- База данных MongoDB

- KMongo

- Kodein

- Docker

1. **Описание выполненного проекта**

Работа над разработкой приложения началась с детальной разработки дизайна приложения, обсуждения подробностей реализации с членами команды и руководителем практики (рисунок 1).

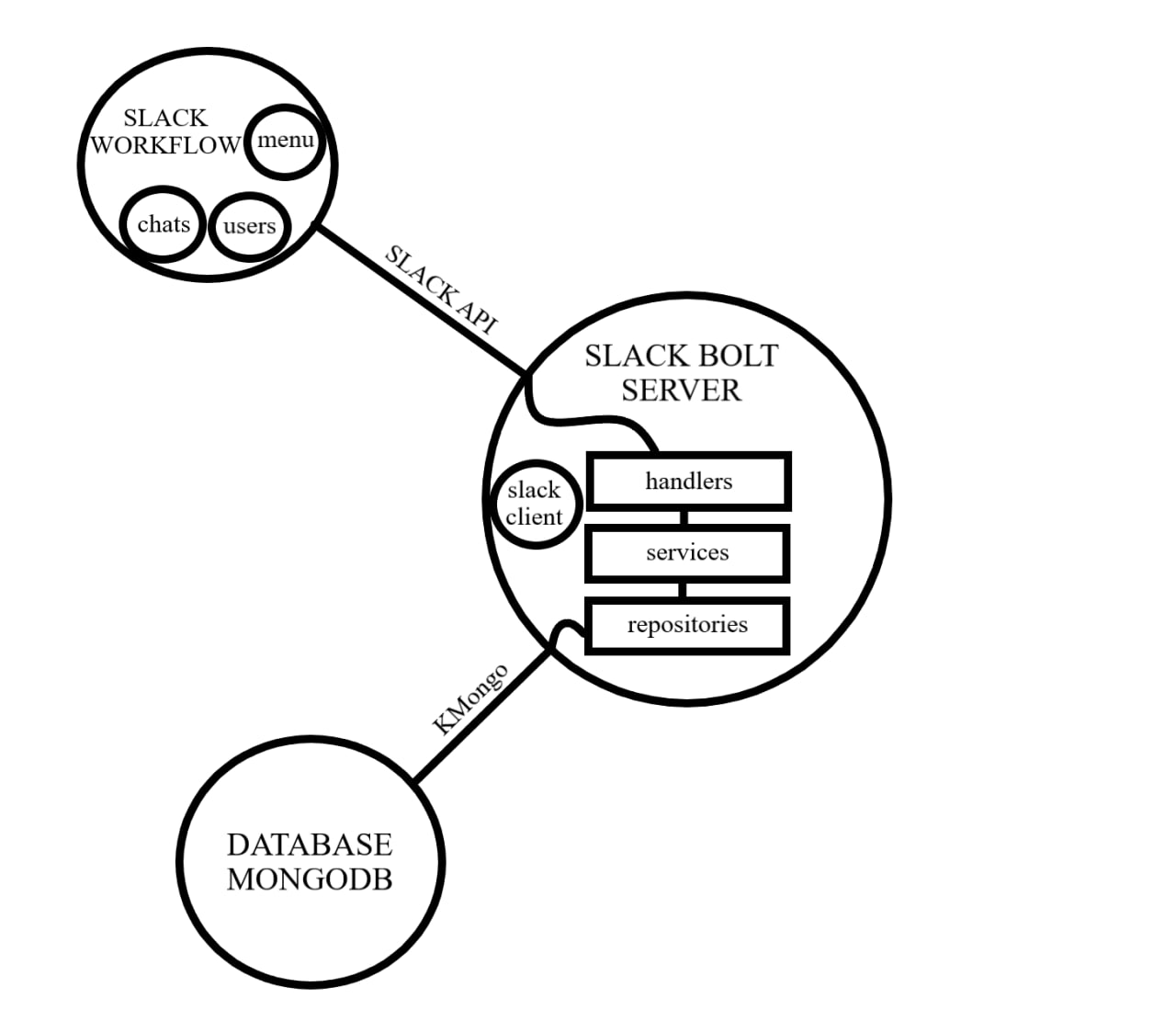


Рисунок 1. Дизайн приложения с основной функциональностью

Slack Bolt Server - центральный объект нашей системы. Использовали Bolt for Java/Kotlin . Bolt для Java — это фреймворк на JVM, который предлагает уровень абстракции для быстрого создания приложений Slack с использованием современных функций платформы. Фреймворк предоставляет приложение с уже написанным сервером, в который только необходимо добавлять endpoint’ы и хендлеры на них. Также в этом приложении предоставляется доступ к клиенту Slack botа, через который можно совершать действия самим Slack ботом в workflow. Например, мы использовали отправления сообщений в каналы или в личные сообщения.

Со стороны Slack бот управляется в специальном разделе сайта slack.api. Именно там можно подписывать бота на какие-то события, на которые он должен реагировать, прописывать команды бота, в обоих случаях необходимо указывать URL сервера бота на который будут высылаться запросы событий или команд. Также здесь можно выдавать права боту и проводить доп. настройки. Эта часть видна пользователю. Здесь было создано визуально удобное меню, для полной конфигурации нашего приложения.

В качестве базы данных используется MongoDB и удобный способ взаимодействия с ней встроенный в Kotlin - Kmongo.

На этапе написания сервера я в основном сталкивалась с проблемой постоянной необходимости переносить части логики на сервер (это было связано как раз таки с интегрированием работы с Youtrack, и подробности возникающих проблем описаны в отчете по учебной практике), что вынуждало немного менять внутреннюю архитектуру приложения.

Кроме написания логики сервера моей задачей в проекте было тестирование. Мы использовали полностью ручное тестирование, где я придумала пул тестов, проверяющих весь основной функционал приложения:

- поведение приложения при появлении новых сообщений в чатах поддержки, а также новых реакций на них.

- поведение приложения при вызове всех доступных команд внешнего взаимодействия с ним (рисунок 2)

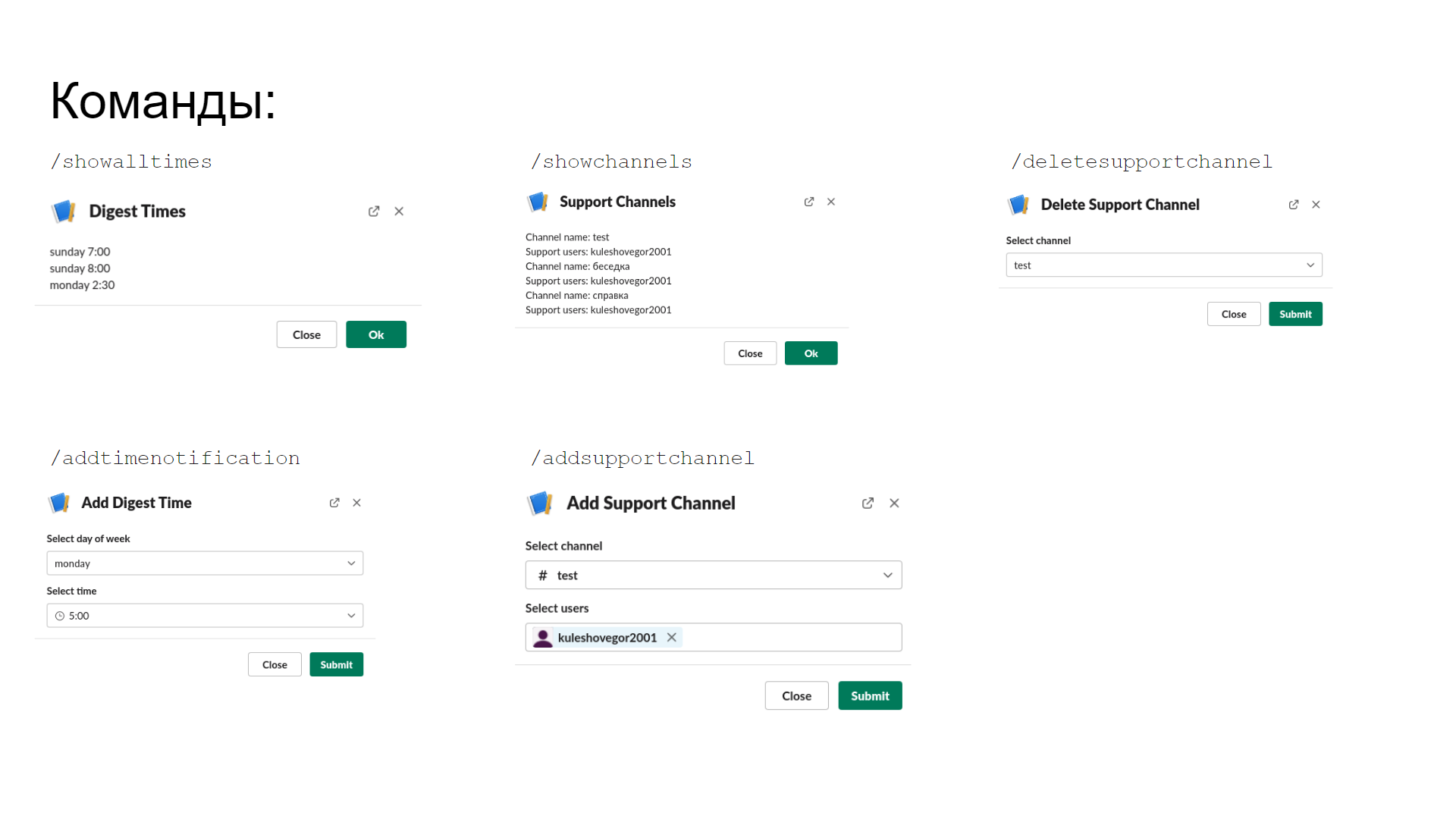


Рисунок 2. Список команд приложения.

- проверка всех возможностей конфигурации приложения:

- тестирование всех функций внешнего интерфейса приложения:

- проверка возможностей настройки роли пользователя приложения (роль admin, и роль user)

- поведение приложения при удалении и добавлении новых пользователей, новых каналов, и новых проектов.

Далее изучались причины неверного или неожиданного поведения приложения на предложенных тестах, проблемы исправлялись, иногда приходилось полностью менять логику кода или вообще искать совершенно другой способ реализации поставленной задачи, после тестирование происходило заново, и так далее по кругу, пока все придуманные тесты не были успешными.

1. **Выводы**

В течение прохождения данной практики были пройдены все этапы разработки прикладного приложения “Smart Reminder”. Были изучены этапы планирования, разработки дизайна, написание компонентов в команде и их последующее совмещение. Также были получены навыки взаимодействия со старшими коллегами по вопросам консультаций и уточнений задач.

1. **Список источников**

1) Документация Slack API // API Slack URL: <https://api.slack.com/docs>

2) База данных MongoDB // MongoDB URL: <https://www.mongodb.com/docs/>

3) Документация KMongo // KMongo URL: <https://litote.org/kmongo/>

4) Документация языка программирования Kotlin // Kotlinlang URL: <https://kotlinlang.org/docs/home.html>

5) Документация Docker // Docker URL: <https://docs.docker.com/>

6) Пример создания Slack приложения // Habr URL: <https://habr.com/ru/company/hh/blog/595837/>

7) Основы Kodein // Habr URL: <https://habr.com/ru/post/431696/>