

Р. А. Ермоленко
(ГГУ имени Ф. Скорины, Гомель)
Науч. рук. **Е. М. Березовская**, канд. физ.-мат. наук, доцент

РАЗРАБОТКА ПРОГРАММНОГО СЕРВИСА ДЛЯ СТОМАТОЛОГИЧЕСКОЙ КЛИНИКИ

Сегодня наблюдается рост популярности частной стоматологической помощи для населения, что приводит к появлению все большего числа программных продуктов, связанных с этой областью. Анализ имеющихся разработок позволяет выделить два направления: простой функционал с интуитивно понятным дизайном для пользователя и автоматизацию нетипичных процессов, которые обычно игнорируются в глобальных приложениях.

Для решения вышеперечисленных проблем разработано и реализовано приложение, которое полностью воссоздает рабочую среду для журнала приемов в стоматологической поликлинике. Это приложение обеспечивает врачей актуальной информацией и позволяет им объективно и детально ознакомиться с предстоящей помощью пациентам.

При разработке приложения важно учесть ряд действий, таких как: авторизация пользователя в системе, профиль пользователя и администрирование, вывод списка приемов, поиск пациентов по ФИО, отображение списка пациентов и приемов, добавление, обновление и удаление пациентов и их приемов, создание раздела с формулой зубов, отслеживание даты и времени посещения, возможность звонка пациентам по контактному телефону. Предусмотрена система разграничение прав доступа между пользователями и администраторами, каждый из которых обладает определенными полномочиями. Реализована удобная навигация по приложению.

В приложении реализована возможность поиска лекарственных средств в аптеках Беларуси с помощью сайта <https://tabletkaby.com>. Посредством этой функции пользователи смогут эффективно искать нужные препараты, сравнивать цены и находить ближайшие аптеки, что значительно облегчит доступ к необходимым медикаментам. Данный сервис необходим для быстрого и удобного доступа к информации в случае, если врач выписал какое-либо лечение или лекарственное назначение.

При разработке использовались React Native, Node.js, Express, база данных MongoDB. Проект был реализован с помощью визуального редактора исходного кода Visual Studio Code, тестирование проводилось на платформе Expo на мобильных устройствах.

А. Д. Жихарева
(ГГУ имени Ф. Скорины, Гомель)
Науч. рук. **М. А. Подалов**, ст. преподаватель

РАЗРАБОТКА МОБИЛЬНОГО ПРИЛОЖЕНИЯ-ОРГАНАЙЗЕРА

Приложения-органайзеры предназначены для содержания личных вещей в определенном владельцем порядке. В большинстве своём приложениями-органайзерами выступают обычные заметки, в которых удобно записывать не только то, где и куда были убраны те или иные вещи, но также держать в порядке свои мысли, планы на ближайшие выходные или список покупок. Но из-за такой многофункциональности часто возникают проблемы с тем, чтобы найти нужную заметку. Для большего удобства были разработаны приложения-ежедневники, но всё же держать в личных заметках записи, которые редко меняются и только занимают место не так удобно, как держать их в отдельном приложении. Поэтому наша цель – создание приложения, в котором пользователи будут хранить списки личных вещей, разложенных по различным полкам и коробкам.

Создаём новый java-интерфейс и импортируем в него библиотеку Dao. Тут же создаём новый публичный интерфейс mainDAO, в котором будут храниться наши SQL-запросы. Начинаем прописывать методы. В отдельном файле создаём новый абстрактный java-класс и в нём создаём нашу базу данных. Прописываем ей версию, название и добавляем проверку на существование при запуске приложения.

Создаём визуальный интерфейс, в который входят все сделанные записи. Они представляют из себя цветной макет, вдоль которого будет располагаться запись, содержание записи, её название и дата создания. Каждая запись кликабельна и доступна для редактирования. Редактирование происходит в отдельном окне, где можно изменить содержание записи или её название, цвет записи задаётся автоматически при создании и остаётся неизменным. Также в приложении реализована функция поиска по записям.

Для защиты приложения от переполнения не используемыми записями мы предусмотрели в нём функцию удаления записей. Реализована она следующим образом. Создаётся элемент PopUp меню, который появляется после длительного нажатия по записи на главном экране. В данном меню имеется кнопка удаления, которая стирает выбранную запись из базы данных и обновляет порядок содержания записей, после чего внизу экрана появится всплывающее уведомление [1] об успешном удалении.

Для отображения работы приложения была создана диаграмма прецедентов. Данная диаграмма отображена на рисунке 1.

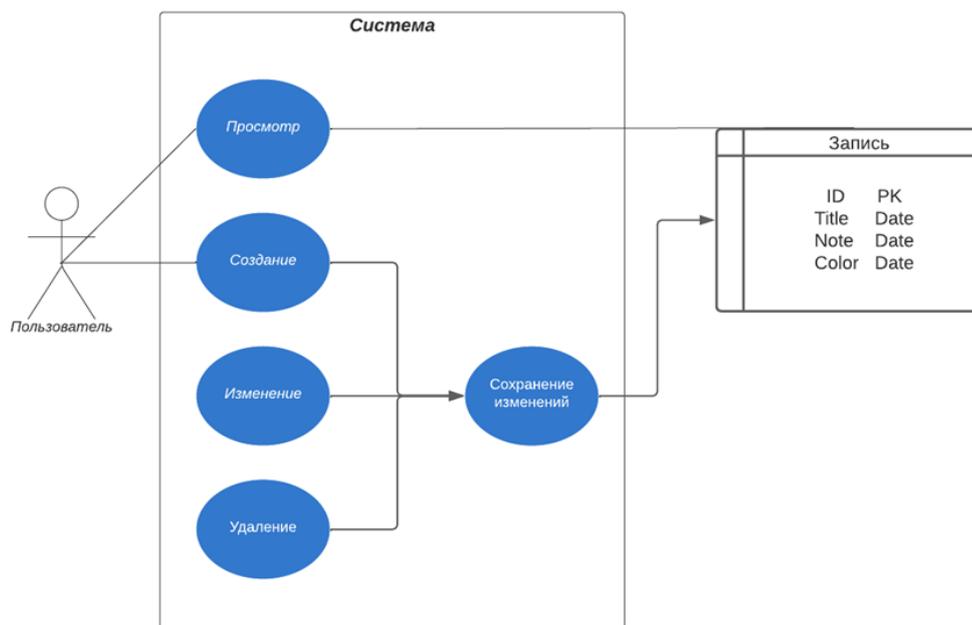


Рисунок 1 – Диаграмма прецедентов

В ходе выполнения данного проекта поэтапно было разработано приложение для организации личных вещей по коробкам, ящикам, полкам и тд. Основываясь на полученном опыте, можно заключить, что разработка мобильных приложений требует не только технической компетенции, но и понимания потребностей конечных пользователей. В ходе проекта были решены технические задачи, также уделено внимание дизайну и удобству использования приложения. Были составлены эскизы приложения со всеми рабочими экранами и проведено сравнение сред разработки. После проведения анализа в качестве лучшей среды была выбрана Android Studio [2] и строго типизированный объектно-ориентированный язык программирования Java [3]. По завершению нескольких этапов разработки были проведены отладка и тестирование приложения, после которых были внесены незначительные корректировки для лучшего функционирования приложения.

Данное приложение предназначено для хранения конкретного рода информации о личных вещах, что значительно упростит поиск необходимых вещей среди множества потенциальных мест их хранения, а также поможет вести учёт количества необходимых предметов в том или ином месте хранения. Все созданные файлы можно найти в удалённом репозитории GitHub [4].

Литература

1. itProger [Электронный ресурс]. – Режим доступа: <https://itproger.com/course/java-android/4> – Дата доступа: 27.11.2023.
2. Wikipedia [Электронный ресурс]. – Режим доступа: https://ru.wikipedia.org/wiki/Android_Studio – Дата доступа: 16.11.2023.
3. Wikipedia [Электронный ресурс]. – Режим доступа: <https://ru.wikipedia.org/wiki/Java> – Дата доступа: 21.11.2023.
4. Github [Электронный ресурс] – Режим доступа: <https://github.com/AZhihareva/CourseWork.git>.

А. А. Жуков

(ГГУ имени Ф. Скорины, Гомель)

Науч. рук. **В. Н. Леванцов**, ст. преподаватель

ОСНОВНЫЕ МЕХАНИЗМЫ ПОДСИСТЕМЫ РЕГУЛИРОВАНИЯ РАБОЧЕЙ СИЛЫ И ЗАРАБОТНОЙ ПЛАТЫ ООО «ГБСОФТ» НА БАЗЕ ПЛАТФОРМЫ 1С

Целью разрабатываемой подсистемы регулирования рабочей силы и заработной платы является реализация возможности эффективного управления и отслеживания движений задач персонала между отделами предприятия и их заработной платы, создания и автоматизации заполнения документов по персоналу.

Разработка подсистемы проводилась на базе типовой конфигурации «1С: Предприятие». В процессе выполнения задания были определены все необходимые элементы для организации учета сотрудников в соответствии с поставленными задачами. В базе данных были созданы справочники, документы и отчеты, которых достаточно для удовлетворения потребностей пользователей системы в надежной и подробной выходной информации. Использование встроенного языка запросов, специально ориентированного на реляционные базы данных, позволило сократить объем работы, который пришлось бы выполнить при использовании какого-либо другого универсального языка программирования.

Для хранения данных были разработаны справочники. В них содержится информация о физических лицах, заработной плате, должностях, учреждениях образования и специальностях. Например, справочник «Физические лица» содержит данные о конкретных людях, а справочник «Сотрудники» включает информацию о принятых на работу сотрудниках предприятия и имеет табличную структуру. Справочник «Задачи» содержит типовые задачи, которые можно назначить сотрудникам.

Некоторые реквизиты справочников имеют ссылочный тип, например Справочник. Ссылка. Должности, что позволяет выбрать необходимую должность из соответствующего справочника, а также Справочник. Ссылка. Задача, что позволяет назначить задачу сотруднику.

В системе имеются два вида документов: «Прием на работу» и «Увольнение». Первый предназначен для оформления приема новых сотрудников на работу, а второй – для увольнения сотрудников.