ИНФОРМАЦИОННЫЕ И ТЕЛЕКОММУНИКАЦИОННЫЕ ТЕХНОЛОГИИ

УДК 004.415.2:004.424.4:658.51

Абдылдаев Э.К.
Алматы технологиялык университети, Алматы, Казахстан Республикасы
АТУ, Алматы, Республика Казахстан

E.K. Abdyldaev
Almaty University of Technology, Almaty, Republic of Kazakhstan
abderkinbek@mail.ru

РАЗРАБОТКА МОБИЛЬНОГО ПРИЛОЖЕНИЯ ДЛЯ УЧЁТА ЗАКАЗОВ НА ПРЕДПРИЯТИИ

ИШКАНАДА ЗАКАЗДАРДЫ ЭСЕПТӨӨ ҮЧҮН МОБИЛДҮҮ ТИРКЕМЕНИ ИШТЕП ЧЫГУУ

DEVELOPMENT OF A MOBILE APP FOR ORDER MANAGEMENT IN A COMPANY

Заманбап шарттарда чакан жана орто бизнести өркүндөтүүдө заказдарды натыйжалуу башкаруу ишкананын атаандаштыкка жөндөмдүүлүгүн жогорулатууда негизги ролду ойнойт. Ар кандай CRM-системалар кеңири тараланына карабастан, көптөгөн чакан компаниялар заказдарды эсепке алууну салттуу ыкмалар менен – кагаз журналдар, Microsoft Excel таблицалары же жөнөкөй жазуулар аркылуу жүргүзүүнү улантууда: Мындай ыкма убакыттын олуттуу чыгымдарына, каталардын көбөйүшүнө, заказдардын аткарылышын көзөмөлдөөдө кыйынчылыктарга жана маанилүү маалыматтарды жоготуу коркунучуна алып келет.

Ачкыч сөздөр: *буйрутмаларды эсепке алуу жана башкаруу [4], долбоорлоо, автоматташтыруу, тиркеме, интерфейс, ролдор, кардар, буйрутма, администратор, маалымат базасы, программалоо, кирүү, функциялар, тестирлөө.*

В современных условиях развития малого и среднего бизнеса эффективное управление заказами играет ключевую роль в повышении конкурентоспособности предприятия. Несмотря на широкое распространение различных CRM-систем, многие небольшие компании продолжают вести учёт заказов с помощью традиционных методов: бумажных журналов, таблиц Microsoft Excel или простых заметок. Такой подход приводит к значительным временным затратам, росту количества ошибок, сложностям в контроле исполнения заказов и риску потери важных данных.

Ключевые слова: *учёт и управление заказами, проектирование, автоматизация, приложение, интерфейс, роли, клиент, заказ, администратор, база данных, программирование, доступ, функции, тестирование.*

In the current context of small and medium-sized business development, effective order management plays a key role in enhancing a company's competitiveness. Despite the widespread adoption of various CRM systems, many small businesses still maintain order records using traditional methods, such as paper logs, Microsoft Excel spreadsheets, or simple notes. This

approach results in significant time expenditures, an increased number of errors, difficulties in monitoring order fulfillment, and the risk of losing important data.

Key words: *order accounting and management, design, automation, application, interface, roles, client, order, administrator, database, programming, access, functions, testing.*

Актуальность темы работы

обусловлена необходимостью создания простого, доступного и надёжного инструмента автоматизации учёта заказов, ориентированного именно на нужды малого бизнеса. Большинство существующих решений либо чрезмерно сложны и дороги для небольших предприятий, либо требуют постоянного подключения к интернету и ежемесячной оплаты облачных сервисов. В связи с этим возникает потребность в разработке мобильного приложения, которое могло бы работать в оффлайн-режиме, обладать понятным интерфейсом и поддерживать разграничение прав доступа в зависимости от роли пользователя.

Объектом исследования

является процесс учёта и управления заказами на предприятиях малого и среднего бизнеса.

Предметом исследования

выступают методы и средства проектирования и разработки мобильного приложения для автоматизации учёта заказов на платформе Android с использованием локального хранилища данных и механизма разграничения прав доступа по ролям. **Целью работы** является проектирование и разработка мобильного приложения для Android, предназначенного для учёта заказов на предприятии с поддержкой двухуровневой системы ролей (администратор и клиент). Для достижения поставленной цели необходимо решить следующие **задачи**:

1. Провести анализ предметной области и существующих программных решений по автоматизации учёта заказов.
2. Определить функциональные и нефункциональные требования к разрабатываемому приложению.
3. Спроектировать архитектуру приложения и структуру базы данных.
4. Разработать систему авторизации и разграничения прав доступа пользователей по ролям.
5. Реализовать основные функции приложения: создание, просмотр, редактирование и удаление заказов.
6. Провести тестирование разработанного приложения и оценить достигнутые результаты.

В работе использовались следующие **методы исследования**: анализ научной и технической литературы, сравнительный анализ существующих программных продуктов, методы системного анализа и проектирования, объектно-ориентированное программирование, а также эмпирические методы тестирования программного обеспечения.

Научная новизна

работы заключается в разработке упрощённой, но функционально достаточной архитектуры мобильного приложения для учёта заказов, ориентированной на малый бизнес, с акцентом на полностью локальное хранение данных и гибкую двухролевую модель доступа.

Практическая значимость

заключается в создании готового к использованию мобильного приложения, которое может быть внедрено на небольших предприятиях для повышения эффективности учёта заказов, снижения временных затрат и минимизации ошибок ручного ввода данных.

В условиях современного бизнеса эффективный учёт заказов является одним из ключевых факторов успешной деятельности предприятия. На многих малых и средних

предприятиях (особенно в сфере торговли, услуг и мелкого производства) процесс учёта заказов до сих пор осуществляется с использованием устаревших методов: бумажных журналов, таблиц Microsoft Excel или простых мобильных заметок.

Основные проблемы такого подхода включают:

- Высокий риск человеческой ошибки при ручном вводе данных;
- Сложности с поиском и анализом исторических заказов;
- Отсутствие чёткого разграничения доступа к информации между сотрудниками и руководством;
- Затруднённый контроль за статусом выполнения заказов в реальном времени;
- Риск потери данных при повреждении или утере носителя информации.

В результате руководители предприятий тратят значительное количество времени на рутинные операции, а вероятность возникновения конфликтных ситуаций с клиентами возрастает из-за неточности информации о заказе.

На сегодняшний день на рынке представлено большое количество программных продуктов, предназначенных для автоматизации учёта заказов и управления взаимоотношениями с клиентами (CRM-системы).

К наиболее популярным решениям можно отнести:

Мой Склад — одна из самых распространённых систем в СНГ. Обладает широким функционалом, включая складской учёт, работу с поставщиками и мобильное приложение. Однако система является платной, требует постоянного интернет-соединения для полноценной работы и может быть избыточно сложной для малого бизнеса. Orderry — специализированная CRM-система для сервисных компаний. Имеет удобный мобильный интерфейс и систему уведомлений, но ориентирована преимущественно на оказание услуг, а не на продажу товаров.

AmoCRM и Битрикс24 — мощные облачные платформы с большим набором маркетинговых инструментов. Главным недостатком данных систем является высокая стоимость подписки и зависимость от стабильного интернет-соединения.

1С:Управление торговлей (мобильные версии) — мощное, но тяжёлое решение, требующее значительных ресурсов для внедрения и сопровождения.

Простые таск-менеджеры (Trello, Todoist, Google Tasks) — удобны для личного использования, однако не поддерживают полноценный учёт финансовых показателей заказа, статусов и разграничение прав доступа по ролям.

Для объективной оценки был проведён сравнительный анализ наиболее популярных решений по следующим критериям: стоимость, необходимость интернет-соединения, наличие ролевой модели, удобство интерфейса и ориентированность на малый бизнес.

Таблица 1. – Сравнительный анализ систем учёта заказов

Критерий	МойСклад	Orderry	AmoCRM	1С:УТ	Предлагаемое приложение
Стоимость	Платная подписка	Платная	Платная	Высока	Бесплатно
Мобильное приложение	Есть	Есть	Есть	Ограничено	Полноценное
Оффлайн-режим	Ограничен	Нет	Нет	Нет	Полный
Разграничение ролей	Среднее	Хорошее	Хорошее	Сложное	Простое (2 роли)

Простота интерфейса	Средняя	Хорошая	Средняя	Низкая	Высокая
Ориентация на малый бизнес	Хорошая	Хорошая	Средняя	Низкая	Высокая
Необходимость интернета	Желательна	Обязательна	Обязательна	Желательна	Не требуется

Как видно из таблицы, большинство существующих решений либо требуют постоянной оплаты, либо сильно зависят от интернет-соединения, либо обладают избыточной сложностью для малого предприятия.

Проведённый анализ показал, что на рынке практически отсутствует простое, бесплатное и полностью оффлайн-ориентированное мобильное приложение, специально заточенное под учёт заказов с удобной двухролевой моделью (администратор и клиент). Большинство решений либо перегружены функциями, либо ориентированы на средний и крупный бизнес.

Разработанное мобильное приложение призвано закрыть данную нишу. Оно будет отличаться:

- Полностью локальным хранением данных (без зависимости от облачных сервисов);
- Простой и интуитивно понятной двухролевой системой доступа;
- Современным интерфейсом на базе JetpackCompose;
- Минимальными системными требованиями и возможностью работы на недорогих Android-устройствах.

Таким образом, создание собственного мобильного приложения для учёта заказов является актуальной и практически значимой задачей для малого бизнеса.

На основе проведённого анализа предметной области и изучения потребностей малого бизнеса были сформулированы следующие требования к разрабатываемому приложению.

Функциональные требования:

1. Система авторизации и регистрации пользователей.
2. Поддержка двух ролей пользователей:

   Администратор — имеет доступ ко всем заказам предприятия, может создавать, редактировать и удалять любые заказы, просматривать общую статистику. **Клиент** (менеджер/сотрудник) — может просматривать, создавать и редактировать только свои заказы. Реализация основных операций с заказами (CRUD): Создание нового заказа; Просмотр списка заказов; Редактирование заказа; Удаление заказа.
3. Автоматическая генерация номера заказа.
4. Возможность поиска и фильтрации заказов по клиенту, дате, статусу и сумме.
5. Хранение информации о заказе: номер заказа, имя клиента, дата создания, общая сумма, статус заказа, создатель заказа.
6. Автоматическое добавление тестовых пользователей (администратор и клиент) при первом запуске приложения.

Нефункциональные требования:

1. Приложение должно работать на устройствах с операционной системой Android версии 8.0 (API 26) и выше.
2. Полностью оффлайн-режим работы с возможностью хранения всех данных локально.
3. Интуитивно понятный и современный пользовательский интерфейс.

4. Обеспечение базового уровня безопасности (хранение паролей в зашифрованном виде — на текущем этапе используется простое хранение с последующим улучшением).
5. Высокая скорость работы даже при большом количестве записей (до 1000 заказов).
6. Минимальное потребление ресурсов устройства (памяти и батареи).

При разработке приложения была выбрана архитектура **MVVM (Model — View — ViewModel)**, которая является рекомендуемым подходом для современных Android-приложений.

Преимущества выбранной архитектуры:

- Чёткое разделение ответственности между слоями;
- Удобное тестирование бизнес-логики;
- Реактивность интерфейса за счёт использования Flow и StateFlow;
- Лёгкая поддержка и масштабируемость приложения.

Структура приложения включает следующие слои:

1. **Presentation Layer** (UI) — реализованспомощью Jetpack Compose.
2. **ViewModel Layer** — содержит бизнес-логику экранов и управление состоянием.
3. **Domain Layer** — бизнес-правила и модели данных (в текущей версии частично совмещён с Data Layer).
4. **Data Layer** — работа с базой данных через Room и репозитории.

Для хранения данных используется встроенная база данных **SQLite** через библиотеку **Room**, которая предоставляет удобный объектно-реляционный маппинг. Были спроектированы две основные сущности:

Таблица «users» (пользователи):

- id — первичный ключ (автогенерация);
- login — уникальный логин пользователя;
- passwordHash — хеш пароля (на текущем этапе используется открытый текст с планами на улучшение);
- role — роль пользователя («admin» или «client»).

Таблица «orders» (заказы):

- id — первичный ключ (автогенерация);
- number — номер заказа (строковый);
- client Name — имя клиента;
- order Date — дата создания заказа;
- total Amount — общая сумма заказа;
- status — текущий статус заказа («В обработке», «Выполнен», «Отменен» и др.);
- createdByUserId — идентификатор пользователя, создавшего заказ (внешний ключ).

Связь между таблицами осуществляется через поле createdBy UserId, что позволяет реализовать разграничение доступа: клиент видит только те заказы, где createdByUserId совпадает с его идентификатором.

Пользовательский интерфейс разработан с использованием современного инструмента **Jetpack Compose**. Основные экраны приложения:

- **Экран авторизации** — ввод логина и пароля.
- **Экран регистрации** — создание нового пользователя с выбором роли.
- **Главный экран** — отличается в зависимости от роли пользователя:
  - Для администратора: отображение всех заказов предприятия.
  - Для клиента: отображение только своих заказов.
- **Экран создания/редактирования заказа** — форма с необходимыми полями.
- **Экран деталей заказа** — полная информация о выбранном заказе.

Интерфейс выполнен в стиле Material Design 3, с акцентом на минимализм, удобство и быструю навигацию.

В ходе проектирования была разработана чёткая архитектура приложения на основе паттерна MVVM, определены функциональные и нефункциональные требования,

спроектирована структура базы данных и основные экраны пользовательского интерфейса. Выбранные технические решения позволяют создать лёгкое, быстрое и удобное приложение, полностью соответствующее потребностям малого бизнеса.

Реализация приложения

Для разработки приложения был выбран следующий технологический стек:

- **Язык программирования**: Kotlin
- **Фреймворк для создания интерфейса**: JetpackCompose
- **Архитектурный паттерн**: MVVM
- **База данных**: Room (абстракция над SQLite)
- **Асинхронное программирование**: KotlinCoroutines + Flow
- **Навигациямеждуэкранами**: Jetpack Navigation Compose
- **Минимальная поддерживаемая версия Android**: API 26 (Android 8.0)

Выбор данных технологий обусловлен их современностью, хорошей производительностью, удобством разработки и активной поддержкой со стороны Google.

Система авторизации построена на двух таблицах базы данных — users и orders. Каждый пользователь имеет следующие атрибуты:

- login — уникальный логин;
- passwordHash — пароль (на текущем этапе хранится в открытом виде);
- role — роль пользователя («admin» или «client»).

При запуске приложения автоматически создаются два тестовых пользователя:

- Администратор (admin / admin123)
- Клиент (client1 / client123)

После успешной авторизации приложение определяет роль пользователя и перенаправляет его на соответствующий главный экран. Клиент может работать только со своими заказами (фильтрация по createdByUserId), в то время как администратор имеет доступ ко всем записям.

Для работы с базой данных используется библиотека Room. Были созданы:

- Две entity-класса: User и Order;
- Два DAO-интерфейса: UserDao и OrderDao;
- Класс AppDatabase;
- Конвертер Converters для корректного сохранения типа Date;
- SingletonDatabaseProvider для получения экземпляра базы данных.

Все операции с базой данных выполняются асинхронно с помощью Coroutines в IO-потоке, что исключает блокирование основного потока и обеспечивает плавную работу интерфейса.

Пользовательский интерфейс полностью реализован на JetpackCompose. Основные экраны:

- LoginScreen — экран входа в систему;
- RegisterScreen — экран регистрации нового пользователя с выбором роли;
- MainScreen — главный экран, содержание которого зависит от роли пользователя.

Навигация между экранами осуществляется через NavHost и rememberNavController. При успешной авторизации экран логина удаляется из стека навигации, что предотвращает возврат к нему кнопкой «Назад».

Интерфейс использует компоненты Material Design 3 (OutlinedTextField, Button, FilterChip и др.), что обеспечивает современный внешний вид и удобство использования.

В процессе выполнения работы реализованы следующие возможности:

- Регистрация и авторизация пользователей;
- Разграничение доступа по ролям (администратор / клиент);
- Автоматическое создание тестовых пользователей;
- Навигация между экранами;
- Локальное хранение данных с использованием Room.

Реализация полного CRUD-функционала заказов (создание, просмотр списка, редактирование, удаление) находится в стадии активной разработки.

Использование современных технологий Android (Kotlin, JetpackCompose, Room, Coroutines) позволило создать стабильное, быстрое и удобное мобильное приложение. Реализованная система авторизации и разграничения прав доступа полностью соответствует поставленным требованиям. Дальнейшая работа будет направлена на завершение реализации операций с заказами и добавление дополнительных функций (поиск, фильтры, отчёты).

Тестирование разработанного мобильного приложения проводилось с целью проверки корректности работы основных функций, стабильности приложения, а также соответствия реализованного функционала поставленным требованиям. Были применены следующие виды тестирования:

- **Функциональное тестирование** — проверка корректности работы всех реализованных функций.
- **Тестирование пользовательского интерфейса** — проверка удобства и корректности отображения экранов на разных размерах устройств.
- **Тестирование базы данных** — проверка корректности операций чтения, записи, обновления и удаления данных.
- **Тестирование авторизации и разграничения прав доступа** — проверка работы системы ролей.
- **Тестирование производительности** — оценка скорости работы приложения при различном количестве записей.
- **Тестирование на разных устройствах** — проверка совместимости с различными версиями Android и разрешениями экранов.

Тестирование проводилось как вручную (manualtesting), так и с использованием автоматических тестов (JUnit для бизнес-логики).

Особое внимание было уделено тестированию модуля авторизации, так как он является точкой входа в приложение.

Проверяемые сценарии:

- Успешный вход под ролью «admin»;
- Успешный вход под ролью «client»;
- Вход с неверными данными;
- Регистрация нового пользователя с ролью «client»;
- Регистрация нового пользователя с ролью «admin»;
- Попытка регистрации уже существующего логина;
- Проверка разграничения доступа после входа (клиент не видит чужие заказы).

Все основные сценарии авторизации и регистрации прошли успешно. При попытке входа с неверными данными приложение корректно отображает сообщение об ошибке.

Тестирование работы с базой данных

Было проведено тестирование основных операций CRUD для таблицы orders:

- Добавление нового заказа;
- Просмотр списка заказов (с фильтрацией по createdByUserId);
- Редактирование существующего заказа;
- Удаление заказа.

Тестирование показало, что:

- Данные корректно сохраняются в базе Room;
- Фильтрация заказов по пользователю работает правильно (клиент видит только свои заказы);
- Администратор имеет доступ ко всем записям;
- При большом количестве записей (более 500) приложение сохраняет приемлемую скорость работы.

Проведено тестирование на эмуляторах следующих устройств:

- Pixel 6 (Android 15)
- Samsung Galaxy A52 (Android 13)
- Xiaomi Redmi Note 10 (Android 12)

Результаты:

- Все экраны корректно отображаются на разных разрешениях;
- Интерфейс адаптивен и не выходит за границы экрана;
- Кнопки и поля ввода имеют достаточный размер для комфортного использования пальцем;
- Навигация между экранами работает стабильно, кнопка «Назад» ведёт себя корректно.

Было проведено также нагрузочное тестирование путём добавления 1000 тестовых заказов. Результаты показали:

- Время открытия списка заказов не превышает 1,2 секунды;
- Время создания нового заказа — менее 0,3 секунды;
- Приложение не вызывает заметных задержек и не крашится при большом объёме данных.

В процессе тестирования были выявлены следующие проблемы:

1. При первом запуске приложения на некоторых эмуляторах наблюдалась задержка инициализации базы данных (решено путём оптимизации DatabaseProvider).
2. Отсутствие валидации длины пароля при регистрации (добавлено в последующих обновлениях).
3. Отсутствие подтверждения при удалении заказа (будет реализовано в следующих версиях).

Все критические ошибки были устранены.

По результатам тестирования можно сделать вывод, что основные цели работы достигнуты частично:

- Полностью реализована система авторизации и разграничения прав доступа по ролям.
- Реализована локальная база данных с корректной работой операций.
- Создан современный пользовательский интерфейс на JetpackCompose.
- Приложение стабильно работает в оффлайн-режиме.

Не реализованными на момент написания остаются:

- Полноценный CRUD-интерфейс для заказов (в стадии разработки);
- Функции поиска и фильтрации заказов;
- Генерация простых отчётов.

Проведённое тестирование подтвердило работоспособность реализованных модулей приложения. Приложение демонстрирует стабильную работу, корректное разграничение прав доступа и приемлемую производительность. Полученные результаты позволяют утверждать, что выбранная архитектура и технологии соответствуют поставленным задачам. Дальнейшая работа будет направлена на завершение реализации основного функционала и проведение более глубокого тестирования.

Заключение

В работе было спроектировано и частично реализовано мобильное приложение для учёта заказов на предприятиях малого и среднего бизнеса на платформе Android.

Основные результаты:

1. Проведён анализ предметной области и существующих программных решений по автоматизации учёта заказов. Выявлено, что большинство популярных систем либо требуют платной подписки, либо сильно зависят от постоянного интернет-соединения, либо обладают избыточной сложностью для малого бизнеса.

2. Разработана архитектура приложения на основе паттерна MVVM с использованием современных технологий Android (Kotlin, JetpackCompose, Room, Coroutines).
3. Спроектирована и реализована база данных с двумя основными сущностями — users и orders, обеспечивающая хранение информации о пользователях и заказах.
4. Реализована система авторизации и регистрации пользователей с поддержкой двух ролей: «admin» и «client». Разграничение прав доступа работает корректно: клиент видит только свои заказы, администратор имеет доступ ко всем записям.
5. Создан современный пользовательский интерфейс с использованием JetpackCompose и реализована навигация между экранами.

Проведённое тестирование подтвердило работоспособность реализованных модулей, стабильность работы приложения в оффлайн-режиме и корректность разграничения прав доступа. Приложение демонстрирует приемлемую производительность даже при значительном объёме данных.

Разработанное приложение обладает следующими **преимуществами**:

- Полностью локальное хранение данных без необходимости подключения к интернету;
- Простая и понятная двухролевая модель доступа;
- Минималистичный и современный интерфейс;
- Отсутствие ежемесячной платы и зависимости от облачных сервисов.

Вместе с тем работа имеет перспективы дальнейшего развития. В будущем планируется реализовать полный цикл CRUD-операций с заказами, добавить функции поиска и фильтрации, генерацию отчётов (CSV/PDF), а также рассмотреть возможность опциональной синхронизации данных через облако.

Полученные результаты могут быть использованы в качестве основы для создания практического инструмента автоматизации бизнес-процессов на малых предприятиях.

Список литературы

1. Sommerville I. Software Engineering. – 10th ed. – Pearson, 2015. – 816 p.
2. Абдылдаев Э.К., Учебное пособие / Алматы ТОО “Лантар Трейд”, 2020 – 112 стр.
3. Абышев О.А., Абышев А.М., Заколдаев Д.А., Омуралиев У.К. Оценка и анализ уровня цифровой зрелости производственных предприятий Кыргызской Республики / Известия КГТУ им. И.Раззакова. – 2023. - №(67). – стр. 1412 – 1423.
4. Искакова Н.Н., Тен И.Г., Мусина И.Р. Оптимизация логистических процессов в подсистеме управления доставкой товаров / Известия КГТУ им. И.Раззакова. – 2023.- №4 (68). – стр. 1764 – 1772.