

«Национальный исследовательский университет
«Высшая школа экономики»

Лицей

Индивидуальная выпускная работа
«Glowws» - агрегатор идей

Выполнил Белозеров Ярослав Артёмович

Научный консультант:
Егоров Дмитрий Сергеевич

Москва 2024

Оглавление

Описание продукта.....	3
Проблемное поле	3
Целевая аудитория	3
Функциональные требования (по проблемной области)	4
Управление идеями и пунктами, как элементами базы знаний	4
Помощь в формулировке и развитии идей.....	4
Аналоги.....	5
Notion.....	5
Obsidian	5
Pinterest.....	5
Google Keep или Apple Заметки	6
Trello	6
Evernote	6
Стек технологий	6
Рефлексия	7
Проблемы и решения.....	7
Дальнейшая судьба проекта.....	8
Приобретённые навыки	9
Альтернативный путь развития продукта	10

Описание продукта

Мой проект – приложение для сбора идей и управления ими. Оно предоставляет удобный интерфейс, чтобы записывать внутренний ход мыслей. Я принял решение создать этот продукт, так как мне нужно было простое и удобное приложение чтобы привести мысли в порядок, а существующие на рынке решения мне не подошли. Они часто перегружены ненужными функциями или наоборот слишком просты. "Glowws" стремится создать баланс между функциональностью и простотой использования. Пользователи смогут не только записывать свои идеи, но и организовывать их в логические цепочки, что значительно упростит процесс креативного мышления. На момент старта проекта это был не только мой первый опыт в Android-разработке, но и первый серьёзный ИТ-проект в целом.

Проблемное поле

Основная проблема, которую решает приложение: сложность использования и малая ориентированность под естественный ход мысли современных приложений для заметок. В проведённом мной опросе более 50% респондентов ответили, что им бывает трудно фокусироваться и формулировать свои идеи. Также, продукты, призванные решить проблему, были описаны как излишне нагруженные и непонятные. Более 50% респондентов не полностью удовлетворены функциями приложения, которым пользуются.

Целевая аудитория

Целевая аудитория продукта – люди, заинтересованные в более простой организации своих мыслей и построении логических цепочек, в том числе, но не ограничиваясь:

- Студенты и школьники
- Работники креативных индустрий
- Создатели каких-либо продуктов (например, на стадии MVP)

Функциональные требования (по проблемной области)

Управление идеями и пунктами, как элементами базы знаний

- **Менеджмент идей**

Пользователь может создавать, удалять и редактировать идеи. Список идей можно фильтровать по релевантности, сортировать по релевантности, алфавиту, времени изменения и создания. Также есть возможность искать текст в идеях.

- **Работа с архивом идей**

Пользователь может помещать идеи в архив, восстанавливать их оттуда, очищать выбранные идеи или все сразу.

- **Работа с пунктами идеи**

Пользователь может создавать, удалять и редактировать пункты идеи. Также есть возможность добавить к пункту метку «Основной», чтобы он отображался в предпросмотре.

- **Настройка приложения под себя**

Помощь в формулировке и развитии идей

- **Добавление, удаление, изменение локальных языковых моделей**

Пользователь может загружать совместимые языковые модели из внутренней памяти устройства, менять их название, удалять, активировать (приводить в рабочее состояние) и деактивировать. Для удалённых моделей доступна опция добавления ключа доступа.

- **Сжатие пункта идеи с помощью ИИ**

В меню пункта пользователю доступна опция «Сжать», которая поможет перефразировать написанный текст более чётким языком, сократив его длину.

- **Продолжение, вывод или заполнение идеи с помощью ИИ**

В меню создания пункта доступны следующие действия ИИ:

- Продолжить – модель продолжает логическую цепочку до места нажатия
- Вывод – модель делает вывод из всех пунктов над местом нажатия
- Заполнить – дополняет логическую связь между двумя пунктами

Если удалённая модель выполняет действие, у пользователя есть возможность остановить выполнение.

Аналоги

Notion

- + Обилие функций
- + Удобная коллаборация
- + Широкий инструментарий для работы с данными
- Есть только онлайн-режим (оффлайн функции ограничены)
- Сильно перегруженный интерфейс (по результатам опроса)
- На текущий момент не работает в России

Obsidian

- + Открытый формат заметок
- + Расширяемость с помощью плагинов
- Высокий порог входа
- Отсутствие встроенной синхронизации

Pinterest

- + Приятный интерфейс
- + Удобная организация сохранённого контента
- Только изображения

- Неудобно добавлять свой контент

Google Keep или Apple Заметки

- + Простота использования и быстрый доступ к заметкам
- Ограниченные возможности организации заметок
- Не подходит для длинных цепочек идей

Trello

- + Визуальная организация задач
- Ограниченные возможности конкретно для редактирования текста
- Не подходит для глубокого анализа идей

Evernote

- + Удобное сохранение контента
- + Большая экосистема
- + Удобный поиск
- Сильно ограниченная бесплатная версия
- Устаревший интерфейс

Стек технологий

Платформа – Android 8+

Архитектура проекта – MVVM, Clean Architecture

Интерфейс приложения – Jetpack Compose, Material You

Локальные данные – SQLite с помощью библиотеки Room, DataStore

Сетевой клиент – Retrofit с сериализацией через Moshi

Внедрение зависимостей – Dagger Hilt

Функционал ИИ – MediaPipe Inference API + Внешние API: GigaChat, OpenRouter

Дополнительно:

- Сервер обратной связи с пересылкой в Telegram (Go: Chi, SQLite, tgbotapi)
 - + Получение списка внешних API

Рефлексия

Проблемы и решения

- **Ограниченность функционала локальных языковых моделей**

При интеграции локальных больших языковых моделей я столкнулся с тем, что модель от Google часто генерировала бессвязный текст, а модель от Microsoft значительно снижала производительность приложения. Это ограничивало возможности по автоматизации процесса формирования идей.

Решение: Я решил добавить возможность использования API больших языковых моделей, что позволило значительно повысить качество генерируемого текста. Хотя это решение привнесло в проект дополнительные сложности, оно обеспечило более высокую производительность и точность генерации идей.

- **Управление настройками приложения**

На этапе разработки мне потребовалась система управления настройками, которая была бы легко расширяемой. Первоначально я планировал создавать отдельные ключи для каждой настройки, но это оказалось неэффективным.

Решение: Я выбрал DataStore для хранения настроек в формате JSON, что позволило мне упростить процесс добавления новых параметров и их валидации. Это решение сделало приложение более гибким и адаптируемым к изменениям в будущем.

- **Обратная связь от пользователей**

Было сложно получить конструктивную обратную связь о приложении от пользователей. Использовать продвинутые инструменты аналитики мне показалось нецелесообразным при текущем масштабе проекта.

Решение: Я создал сервер обратной связи с пересылкой сообщений в Telegram, который позволяет пользователям оставлять отзывы

непосредственно через приложение. Это значительно упростило процесс получения отзывов и улучшило взаимодействие с аудиторией.

Дальнейшая судьба проекта

Конечно, в любом продукте есть простор для усовершенствования. Некоторые функции, которые, как я считаю, будут уместны для реализации в будущем:

- Предпросмотр ссылок в пункте – пользователи смогут видеть и взаимодействовать с предпросмотром ссылок, добавленных в пункт.
- Добавление в пункт аудио, видео, документов и возможности рисования – медиаконтент поставит продукт ближе по функциям к конкурентам и позволит пользователям более точно выражать мысли. Также это поможет централизовать всю нужную пользователю информацию в приложении, без надобности из него выходить. В перспективе также обработка этих данных с помощью ИИ.
- Сетевые функции – синхронизация между устройствами, резервное копирование в облако.
- Система версий идеи – таймлайн изменений идеи и переход к какой-либо версии будут доступны пользователю.
- Интеграция с другими сервисами – например, резервное копирование в облачные сервисы. Также возможна синхронизация напоминаний с календарями и прочими сервисами.
- Система напоминаний – возможность добавить уведомление к идее, которое придёт в назначенное время и напомнит о чём-либо важном.
- Поддержка функций коллаборации – возможность создавать совместные идеи и работать над ними одновременно.
- Экспорт хранилища в различные форматы – открытость базы данных к выгрузке и отдельной обработке.

Приобретённые навыки

- **Основные принципы разработки приложений для Android**, включая использование современных инструментов и библиотек, таких как Jetpack Compose, DataStore, Kotlin Coroutines/Flows. Эти знания открывают передо мной возможности для дальнейшего изучения мобильной разработки и создания собственных приложений, что может стать основой для будущей карьеры в ИТ.
- **Базовые навыки работы с базами данных** на примере SQLite. Понимание механизмов хранения и получения данных — один из ключевых аспектов разработки в целом.
- **Использование больших языковых моделей** в продукте дало мне представление о том, как интегрировать искусственный интеллект в приложения. Это направление является одним из самых перспективных в ИТ на данный момент, и опыт в нём может значительно повысить мою конкурентоспособность на рынке вакансий.
- **Способность планировать и организовывать процесс разработки** от идеи до реализации. Это включает в себя определение функциональных требований, создание прототипов и тестирование. Эти навыки полезны не только в разработке, но и в любом другом проекте, требующем системного подхода.
- **Умение самостоятельно учиться и искать информацию**, что является одним из важнейших навыков в современном мире.

Альтернативный путь развития продукта

Конечно, некоторые аспекты продукта можно было реализовать по-другому:

- **Вместо фиксированного значения приоритета добавить идеям теги.**

Так у пользователя будет больше свободы в организации. Например, к одной идее можно было бы прикрепить несколько тегов, а потом осуществлять по ним фильтрацию

- **Изначально построить серверную архитектуру** и не опираться на локальную базу данных. Такое решение, как мне кажется, было бы более популярным и улучшило бы опыт использования.

- **Добавить подсказки для новых пользователей.**

Разработка системы встроенных подсказок и туториалов, которые помогали бы пользователям быстро освоиться с функционалом приложения. Например, при первом запуске можно было бы предложить интерактивный тур по основным функциям (onboarding).