

Пермский филиал федерального государственного
автономного образовательного учреждения высшего образования
«Национальный исследовательский университет
«Высшая школа экономики»

Факультет социально-экономических и компьютерных наук

Князев Арсений Алексеевич

РАЗРАБОТКА PWA ДЛЯ ПОИСКА ПОПУТЧИКОВ В ПУТЕШЕСТВИЯ

Курсовая работа

студента образовательной программы «Разработка информационных систем
для бизнеса» по направлению подготовки 09.03.04 Программная инженерия

Руководитель:

Старший преподаватель кафедры ИТБ

Волков С.А.

Пермь, 2024 год

Содержание

Введение	3
1 Анализ предметной области для формирования требований	4
1.1 Обоснование актуальности и практической значимости	4
1.2 Интервью с пользователями	6
1.3 Анализ конкурентов	6
1.4 Макетирование интерфейса	8
1.5 Выбор средств и технологий разработки	9
2 Проектирование базы данных	11
3 Разработка PWA	12
Список литературы	13
Приложение А. Диаграмма User Flow	14

Введение

В повседневной жизни люди всё больше и больше начали полагаться на общение и знакомства через социальные сети, мессенджеры и другие интернет ресурсы [1]. Поэтому и поиск новых знакомств также перетекает в интернет.

Так-же в России ежегодно растет доля внутреннего туризма [2].

Именно эти тренды были основополагающими при выборе темы моего прикладного проекта. Мной было решено совместить эти два тренда в одну идею. Этой идеей стало веб-приложение для поиска попутчиков для путешествий.

Такое приложение подстегнет людей, у которых нет знакомых, с которыми можно отправиться в путешествие, к путешествиям, а также позволит найти новые знакомства.

Объектом автоматизации являются человеческие отношения.

Цель работы – создание информационной системы обеспечивающей пользователям удобный инструмент для поиска попутчиков, с целью совместных путешествий.

1 Анализ предметной области для формирования требований

Анализ решено было разбить на 5 частей:

1. Подтверждение практической значимости и надобности в продукте.
2. Интервью с предполагаемыми пользователями и формирование сценариев взаимодействия.
3. Анализ конкурентов.
4. Макетирование интерфейса.
5. Выбор средств и инструментов для разработки.

Все этапы анализа должны проходить поочередно, потому что каждый последующий этап зависит от результата предыдущего.

1.1 Обоснование актуальности и практической значимости

Анализ было решено начать с подтверждения надобности пользователей в реализации продукта. Для этого воспользовался сервисами Google Trends и Яндекс ВордСтат, которые предлагают получить статистику поисковых запросов по ключевым словам.

Результаты запроса «Поиск попутчиков на отдых» представлены на рисунке 1.1.

Период ↓	Число запросов
март 2023	5 079
апрель 2023	4 774
май 2023	6 903
июнь 2023	8 752
июль 2023	9 917
август 2023	9 241
сентябрь 2023	6 515
октябрь 2023	4 514
ноябрь 2023	4 113
декабрь 2023	3 833
январь 2024	4 761
февраль 2024	4 155
март 2024	4 493

Рисунок 1.1 – Результаты Яндекс ВордСтат

Google Trends не дает доступа к информации о количестве запросов, поэтому его данные не были учтены в анализе, так как не представляется возможным как-либо адекватно интерпретировать данные полученные от этого сервиса. Яндекс ВордСтат в свою очередь предоставляет полные метрики по запросам.

В среднем количество запросов в месяц составляет около 6 тысяч, самые пиковые значения приходятся на июнь, июль и август, периоды, когда люди больше отправляются в путешествия.

Из полученных данных мы понимаем, что на рынке существует потребность в подобном продукте и люди регулярно ищут подобные сервисы.

Еще одним подтверждением актуальности и практической значимости будущего продукта, являются отзывы пользователей похожих продуктов на площадках Google Play и App Store. Отзывы пользователей не только демонстрируют потребность в подобном продукте, но также показывают, что существующие продукты не закрывают потребности пользователей в связи со своей недоработанностью, малой функциональностью и плохой работоспособностью. Комментарии пользователей приложений конкурентов представлены на рисунках с 1.2 по 1.7.



Рисунок 1.2 – Комментарий пользователя

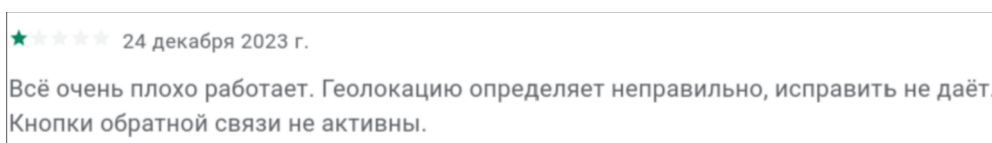


Рисунок 1.3 – Комментарий пользователя

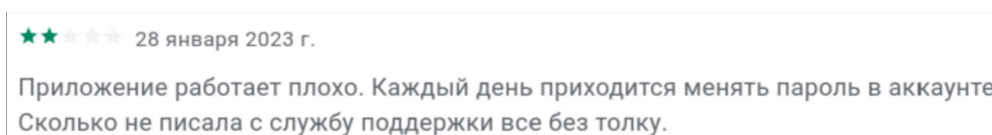


Рисунок 1.4 – Комментарий пользователя

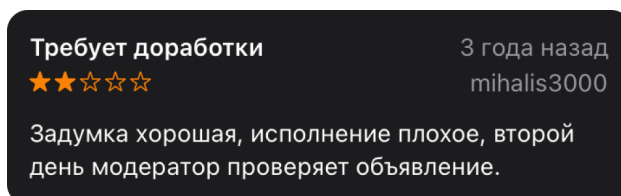


Рисунок 1.5 – Комментарий пользователя

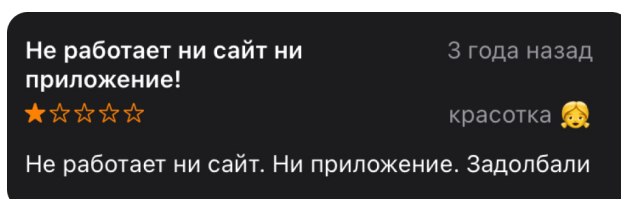


Рисунок 1.6 – Комментарий пользователя

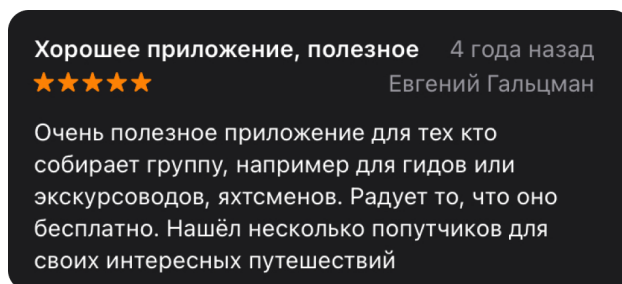


Рисунок 1.7 – Комментарий пользователя

1.2 Интервью с пользователями

Далее были проведены интервью с потенциальными пользователями будущего приложения. На основе проведенных интервью были сформированы сценарии взаимодействия с приложением на основе фреймворка Jobs To Be Done. Сценарии взаимодействия представлены в таблице 1.

Таблица 1 – Jobs To Be Done

Ситуация	Мотивация	Желаемый результат
Когда я хочу отправиться в путешествие в компании, но не имею человека, который сможет поехать со мной	Я хочу найти человека или компанию, которые бы разделяли мои интересы и ценности	Чтобы поездка была наполнена положительными эмоциями, и я нашел новые знакомства
Когда я хочу отправиться в путешествие в компании, но не имею человека, который сможет поехать со мной	Я хочу найти человека или компанию, которые бы разделяли мои интересы и ценности в современном формате и не просматривать кучу странных объявлений	Чтобы я быстро и весело нашел себе попутчика
Когда я хочу поехать в путешествие, но не имею большого бюджета для него	Я хочу найти человека или компанию, с которыми можно разделить часть расходов	Чтобы путешествие получилось бюджетным, но еще более веселым и запоминающимся

По итогам проведенных интервью были выделены 3 задачи, которые пользователи хотят выполнить при помощи приложения по поиску попутчиков в путешествия. Эти задачи помогут смоделировать интерфейс приложения максимально лаконичным и удобным именно под нужды пользователей.

1.3 Анализ конкурентов

Следующим этапом стал анализ конкурентов. Для этого было выбрано 6 приложений прямых и косвенных конкурентов, были выделены все экраны для того, чтобы

проанализировать функционал имеющийся в этих приложениях. Экраны приложений конкурентов представлены на рисунке 1.8.

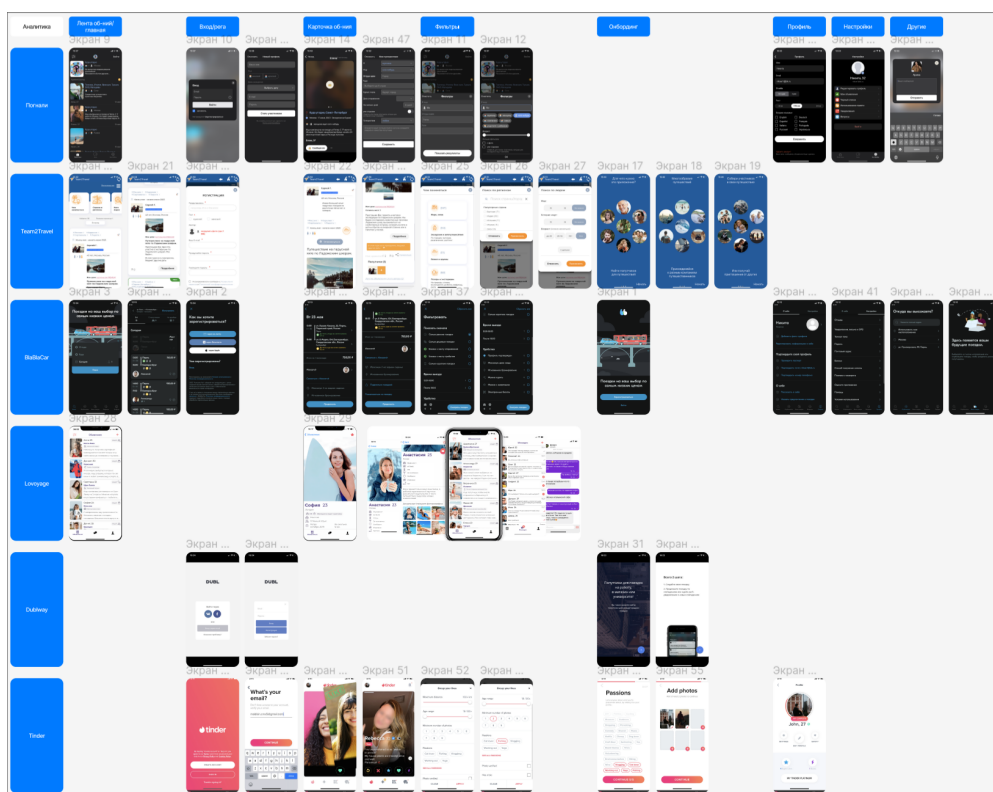


Рисунок 1.8 – Экраны прямых и косвенных конкурентов

Приложения «Погнали», «Team2Travel» и «Lovoyage» имеют схожие экраны главной страницы, она выглядит как доска объявлений со списком всех доступных пользователей. В остальном экраны приложений схожи и обладают похожим функционалом.

Решение с доской объявлений не самое подходящее для такого приложения, так как мы живем в «эпоху рекомендательных алгоритмов», где буквально в каждом приложении или на каждом сайте есть система рекомендаций на основе данных о пользователе, в пример можно привести «TikTok», не только одну из самых популярных, но также самую быстро растущую социальную сеть на данный момент [3], которая поставила рекомендательные алгоритмы во главу приложения, что стало одной из ключевых причин его успеха.

Также в пример можно привести приложение для знакомств «Tinder», которое тоже обладает большой популярностью в своей сфере. Там рекомендательные алгоритмы «решают» какой пользователь попадется вам на основе статистики и метрик.

В нашем же случае мы возьмем модель приложения «Tinder», так как оно доказало эффективность такого решения своей популярностью. Пользователь будет указывать свои интересы и цели на поездку, а уже на основе этих данных пользователю будут предоставляться рекомендуемые кандидаты.

1.4 Макетирование интерфейса

Далее была сформирована схема user flow, см. ПРИЛОЖЕНИЕ А, ней изображен путь пользователя по приложению. Данная схема позволит спроектировать дизайн приложения, не забыв учесть какие-либо функции, а также позволит наглядно изобразить архитектуру будущего приложения.

Итогом проведенного анализа, стал макет приложения реализованный при помощи инструмента Figma. Общий вид спроектированных страниц представлен на рисунке 1.9.

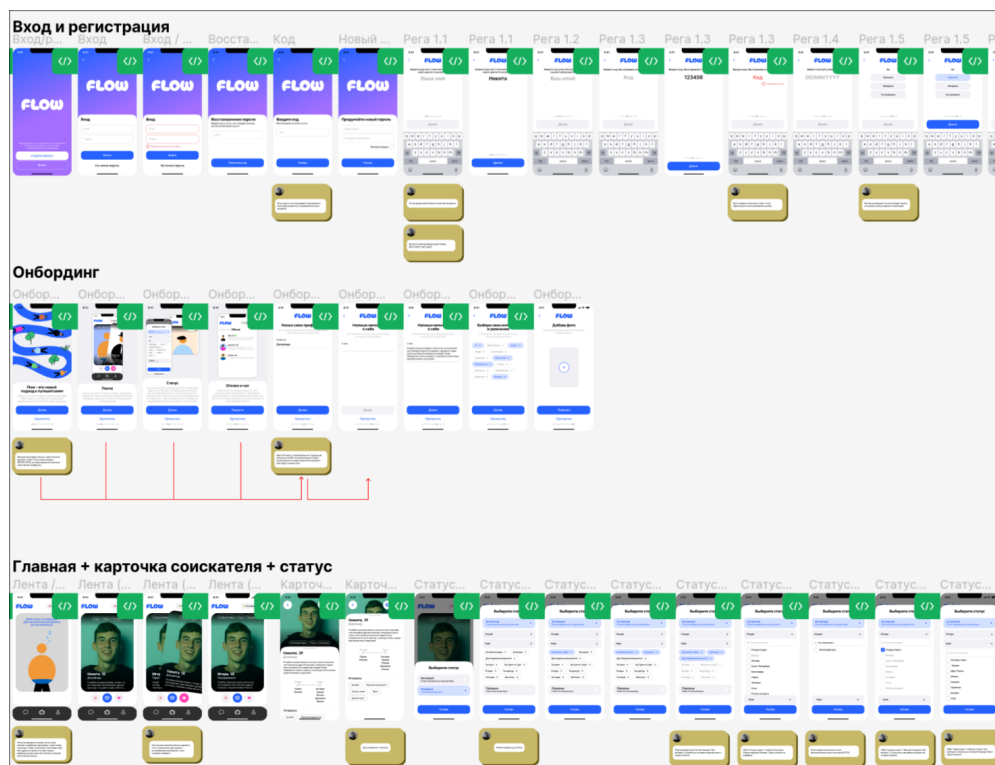


Рисунок 1.9 – Готовые страницы в Figma

Были смоделированы страницы входа и регистрации, главная, чат, профиль пользователя, и модальные окна изменения интересов и целей поездки. Примеры страниц представлены на рисунке 1.10.

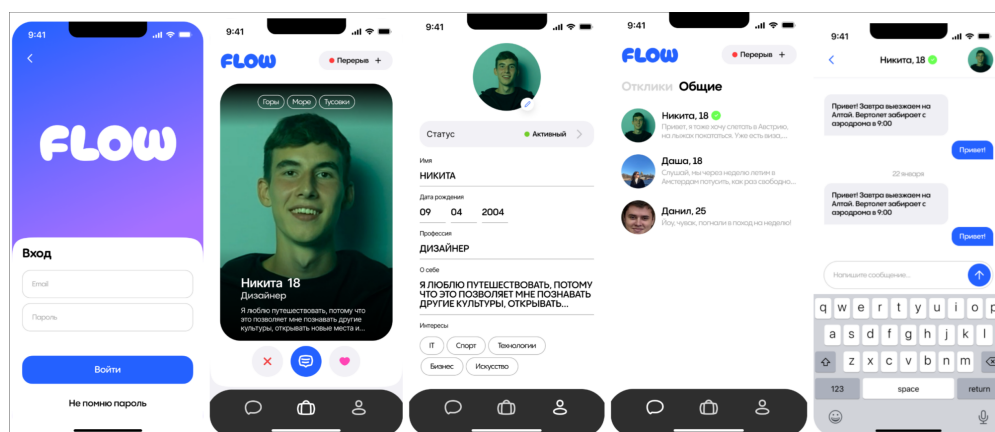


Рисунок 1.10 – Экраны приложения

1.5 Выбор средств и технологий разработки

Так как пользователи мобильных устройств составляют 80% времени проведенного в социальных сетях [4], то было принято решение создать веб-приложение.

Для упрощения разработки под все типы мобильных устройств, решено использовать технологию Progressive Web App (далее PWA). Технология была представлена Google для упрощения создания мультиплатформенных приложений. Технология предоставляет набор инструментов, который позволяет превратить сайт, в нативное приложение с некоторыми оговорками [5].

Для того, чтобы выбрать инструмент для создания клиентской части приложения была составлена сравнительная таблица с 4-мя кандидатами, представленная под номером 2.

Таблица 2 – Сравнение фреймворков

Критерий	React	Vue	Angular	Svelte
Сообщество	Большое	Среднее	Среднее	Маленькое
Производительность	Высокая	Высокая	Умеренная	Очень высокая
Управление состояниями	Стороннее	Встроенное	Стороннее	Встроенное
Стилизация компонентов	Сторонняя	Сторонняя	Сторонняя	Встроенная
Простота изучения	Умеренно	Легко	Сложно	Легко
Количество сторонних библиотек	Огромное	Среднее	Среднее	Малое

Следуя данным из таблицы, одним из самых лучших инструментов оказывается «Svelte», он быстрый, имеет большое количество встроенных функций, которые в других фреймворках требуют сторонние пакеты для их работы. Но, есть небольшая проблема, маленькое сообщество, что может вылиться в невозможность найти решение какой-либо проблемы и малое количество сторонних библиотек, что может привести к проблемам с написанием функционала. Но есть пара нюансов, которые поменяют взгляд на две эти проблемы.

Первое, Svelte имеет интерактивное руководство, которое проводит пользователя «за ручку» и показывает все возможности данного инструмента, начиная с чего-то простого, как переменные и реактивность и заканчивая «Context API» и другим.

Вторая проблема также, по своей сути не является проблемой, Svelte по умолчанию предоставляет все нужные инструменты для разработки: встроенное хранилище состояний, стилизация компонентов, роутинг(при помощи SvelteKIT). Говоря проще, другим фреймворкам требуются сторонние библиотеки для того, чтобы полноценно использоваться в проекте, а Svelte нет [6].

Учитывая всё вышесказанное, в проекте будет использоваться Svelte, а точнее SvelteKIT, фреймворк предоставляющий дополнительные инструменты для Svelte, такие как SSR, Роутинг и многое другое.

Для серверной части было решено ограничиться знакомыми языками программирования, поэтому выбор был из фреймворков на Python и ASP.NET Core. В своей предыдущей работе я использовал ASP.NET Core, поэтому в этот раз было решено попробовать что-то новое, а значит вариантов стало еще меньше.

Самыми популярными Backend фреймворками на Python являются: FastAPI, Flask, Django. Здесь выбор был прост, было решено взять самый быстрый фреймворк, так как скорость работы для backend'а является основным показателем.

На рисунке 1.11 продемонстрировано сравнение скоростей разных фреймворков, на основе на количестве ответов в секунду при 20 запросах на 1 запрос.

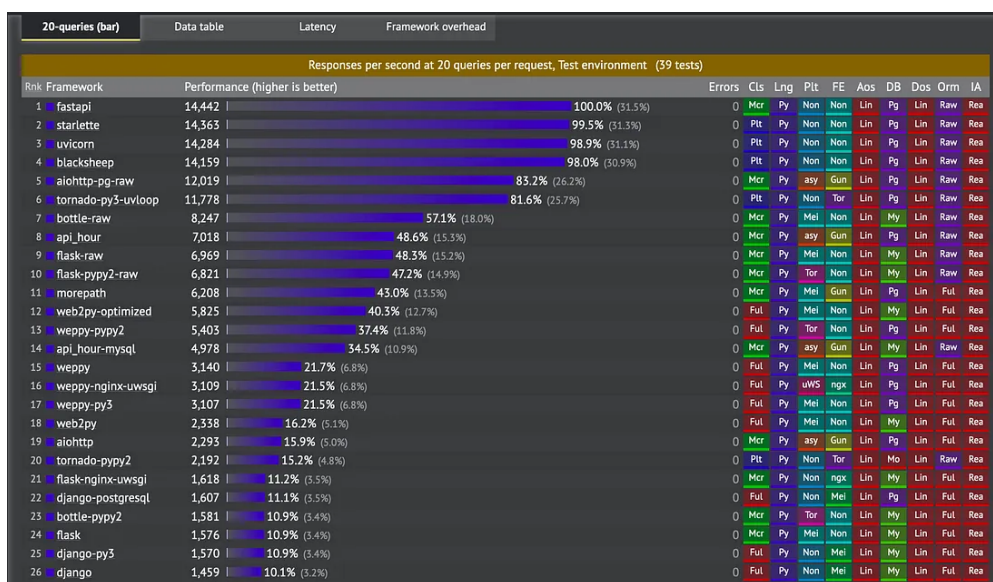


Рисунок 1.11 – Сравнение производительности backend фреймворков

Как мы видим, fastapi является самым быстрым, поэтому в проекте будет использоваться он.

2 Проектирование базы данных

3 Разработка PWA

Список литературы

1. *We Are Social*. DIGITAL 2022: ANOTHER YEAR OF BUMPER GROWTH. — URL: <https://wearesocial.com/uk/blog/2022/01/digital-2022-another-year-of-bumper-growth-2> (дата обр. 30.03.2024).
2. *Ведомости*. Внутренний туризм в России за год увеличился до рекордных 75 млн поездок. — URL: <https://www.vedomosti.ru/society/articles/2024/01/22/1016071-vnutrennii-turizm-v-rf-za-god-velichilsya> (дата обр. 30.03.2024).
3. *Statista Advertising and Media Outlook*. The Rapid Rise of TikTok. — URL: <https://www.statista.com/chart/28412/social-media-users-by-network-am/> (дата обр. 01.04.2024).
4. *Smart Insights and Marketing Land*. 80% of social media browsing is on mobile devices. — URL: <https://techjury.net/blog/mobile-vs-desktop-usage> (дата обр. 30.03.2024).
5. *LePage P., Richard S.* What are Progressive Web Apps? — URL: <https://web.dev/articles/what-are-pwas> (дата обр. 30.03.2024).
6. *roguegpu*. SvelteJS: My ecosystem is bigger than yours. — URL: <https://hackmd.io/@roguegpu/r1RKQMdt3> (дата обр. 01.04.2024).

ПРИЛОЖЕНИЕ А

Диаграмма User Flow

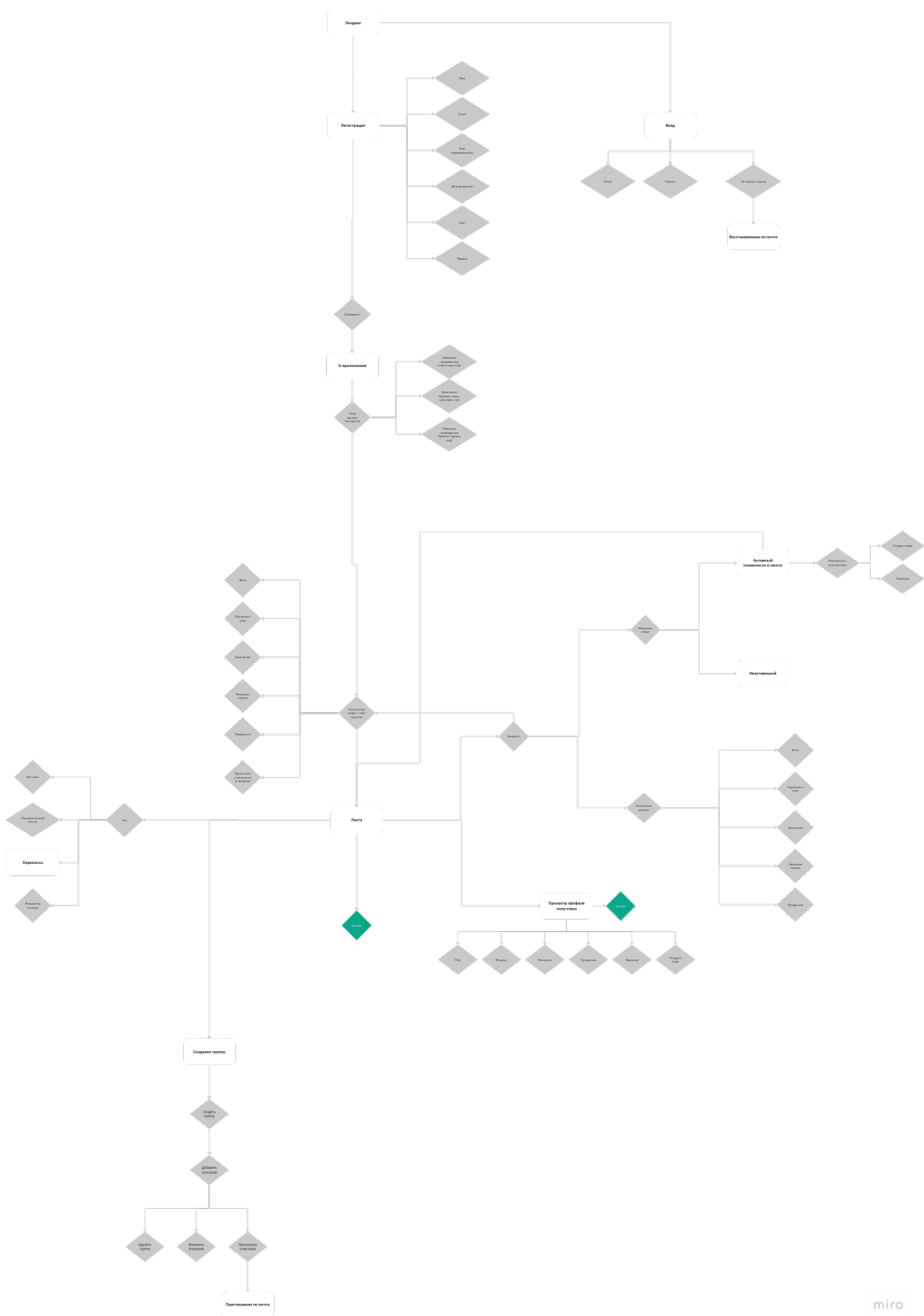


Рисунок А.1 – Диаграмма User Flow