

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РЕСПУБЛИКИ КАЗАХСТАН**

**Казахский национальный исследовательский технический университет
имени К.И. Сатпаева**

Институт автоматизации и информационных технологий

Кафедра «Программная инженерия»

Тоқтар Анель Мұхтарқызы

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА
к дипломному проекту

Образовательная программа 6B06102 – Computer Science

Алматы 2025

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РЕСПУБЛИКИ
КАЗАХСТАН

Казахский национальный исследовательский технический университет имени
К.И.Сатпаева

Институт автоматизации и информационных технологий

Кафедра «Программная инженерия»

ДОПУЩЕН К ЗАЩИТЕ

Заведующий кафедрой ПИ

канд. тех. наук, ассоц. профессор

Ф.Н. Абдолдина

“ 30 ” 05 2025 г.

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к дипломному проекту

На тему: «Разработка мобильного-приложения для контроля личных финансов»

Образовательная программа: 6B06102 Computer Science

Выполнила

Токтар А.М

Рецензент

PhD, заведующего кафедрой
«Computer Technology and
CyberSecurity» МУИТ

“ 05 ” 05 2025 г.
Муханов С.Б.

Научный руководитель

PhD, Ассоциированный профессор
Чинибаев Е.Г

“ 07 ” 06 2025 г.

Алматы 2025

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РЕСПУБЛИКИ
КАЗАХСТАН

Казахский национальный исследовательский технический университет имени
К.И.Сатпаева

Институт автоматизации и информационных технологий

Кафедра «Программная инженерия»

Заведующий кафедрой ПИ канд. тех.
наук, асс. профессор
Ф.Н. Абдолдина
«30» 05 2025 г.



ЗАДАНИЕ

на выполнение дипломного проекта

Обучающемуся: Токтар Анель Мұхтарқызы

Тема: Разработка мобильного-приложения для контроля личных финансов

Утверждена приказом проректора по академической работе:

№ 1804-90

	от « 25 » 11 2024 г.
Срок сдачи законченного проекта	« 25 » 05 2025 г.

Исходные данные к дипломному проекту:

- А) Анализ предметной области, анализ аналогичных проектов;
- Б) Разработка технического задания;
- В) Разработка архитектуры ПО;
- Г) Проектирование и разработка понятного и интерактивного дизайна;
- Д) Разработка и тестирование ПО;

Перечень подлежащих разработке в дипломном проекте вопросов: (с точным указанием обязательных чертежей): представлены 16 слайда презентации.

Рекомендуемая основная литература: из 38 наименований.

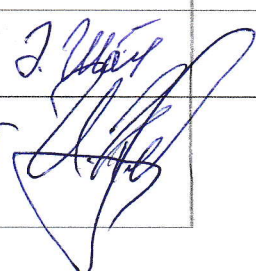
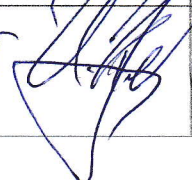
ГРАФИК

подготовки дипломного проекта

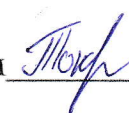
Наименование разделов, перечень разрабатываемых вопросов	Сроки представления научному руководителю и консультантам	Примечание
1. Анализ предметной области, разработка технического задания	15.02.2025	Выполнено
2. Выбор технологий для разработки	04.03.2025	Выполнено
3. Разработка логики взаимодействия	20.03.2025	Выполнено
4. Разработка функционала системы	25.03.2025	Выполнено
5. Тестирование и оптимизация	22.04.2025	Выполнено
6. Написание пояснительной записки к дипломному проекту	12.05.2025	Выполнено

Подписи

консультантов и нормоконтролера на законченную дипломный проект
с указанием относящихся к ним разделов проекта

Наименования разделов	Консультанты, И.О.Ф. (уч. степень, звание)	Дата подписания	Подпись	Оценка
Программное обеспечение	Шаханов Әділ Рауанұлы	30.05.2025		
Нормоконтролер	Имаматдинова Камила Фархадкызы	29.05.2025		

Научный руководитель  Чинибаев Е.Г.

Задание приняла к исполнению обучающаяся  Токтар А.М.

Дата « 25 » май 2025 г.

АҢДАТПА

Бұл бітіру жұмысы жеке қаржыны басқаруға арналған мобильді қосымшаны әзірлеуге бағытталған. Жобаның мақсаты – пайдаланушының табысын, шығындарын, жинақтары мен қаржылық мақсаттарын басқаруды жеңілдететін ақпараттық жүйе құру.

Жұмыс барысында қолданыстағы қосымшаларға талдау жасалып, функционалдық талаптар анықталып, жүйенің архитектурасы жобаланды және негізгі модульдер (операцияларды тіркеу, аналитикалық диаграммалар, еске салғыштар, мақсаттар) жүзеге асырылды. Жоба Django фреймворкі негізінде жүзеге асырылды.

Зерттеу нәтижелері қаржылық технологиялар мен жеке қаржылық жоспарлауды цифрландыру саласында қолдануға болады.

Түйінді сөздер: жеке қаржы, мобильді қосымша, Django, қаржылық жоспарлау, пайдаланушы интерфейсі.

АННОТАЦИЯ

В данной выпускной квалификационной работе рассматривается разработка мобильного приложения для контроля личных финансов. Целью проекта является создание информационной системы, которая позволяет пользователю эффективно управлять доходами, расходами, накоплениями и финансовыми целями с помощью мобильного устройства.

В процессе работы был проведён анализ существующих решений, определены функциональные требования, спроектирована архитектура системы и реализованы ключевые модули, включая регистрацию операций, аналитические графики, напоминания и цели. Разработка велась с использованием фреймворка Django.

Полученные результаты могут быть применены в области финансовых технологий и цифровой трансформации личного финансового планирования.

Ключевые слова

: личные финансы, мобильное приложение, Django, финансовое планирование, пользовательский интерфейс.

.

ABSTRACT

This thesis project presents the development of a mobile application for personal finance control. The aim of the project is to design an information system that enables users to effectively manage income, expenses, savings, and financial goals via a mobile interface.

The work includes analysis of existing solutions, identification of functional requirements, system architecture design, and implementation of main modules such as transaction logging, analytical charts, reminders, and goal tracking. The application is developed using the Django framework.

The results can be applied in the field of fintech and digital personal financial planning.

Keywords: personal finance, mobile application, Django, financial planning, user interface.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	5
1 Теоретические основы учета и контроля личных финансов	7
1.1 Понятие и значение управления личными финансами.....	7
1.2 Современные подходы к финансовому планированию.....	10
1.3 Анализ существующих мобильных приложений для учета финансов.....	14
2 Проектирование мобильного приложения для контроля личных финансов...	24
2.1 Определение функциональных требований.....	24
2.2 Архитектура приложения	29
2.3 Пользовательский интерфейс и UX-дизайн.....	37
2.4 Выбор инструментов и технологий разработки	40
3 Реализация и тестирование мобильного приложения	44
3.1 Этапы разработки	44
3.2 Реализация основных функций	50
3.2.1 Учёт доходов и расходов	51
3.2.2 Финансовая аналитика	53
3.2.3 Напоминания	55
3.2.4 Финансовые цели и планы	56
ЗАКЛЮЧЕНИЕ	59

ВВЕДЕНИЕ

Бурное развитие информационных технологий оказывает значительное влияние на многие сферы современного общества, в том числе на сферу управления личными финансами. Эффективное проведение финансовых операций, правильное планирование и контроль личных денежных средств играют важную роль в жизни каждого человека. В настоящее время актуальными становятся вопросы повышения финансовой грамотности населения и оптимального управления доходами и расходами. Особенно важно то, что инструменты управления личными финансами с помощью мобильных приложений становятся неотъемлемой частью повседневной жизни.

Актуальность дипломной работы связана с растущей значимостью автоматизации процесса управления финансовыми операциями частных лиц на фоне развития финансовых технологий. В настоящее время на территории Республики Казахстан широко используются мобильные приложения для управления личными финансами. Однако большинство из них не в полной мере соответствует реальным потребностям пользователей или обладают ограниченным функционалом. В связи с этим возникает необходимость разработки доступного, простого и функционального мобильного приложения с применением современных технологий.

Цель дипломной работы — разработать информационную систему мобильного приложения, предназначенного для управления финансовыми операциями частного лица, и реализовать её на практике.

Объект исследования — процесс управления финансовыми действиями частного лица.

Предмет исследования — мобильное приложение для управления личными финансами.

Задачи работы:

- Теоретическое исследование понятия управления личными финансами;
- Рассмотрение современных методов финансового планирования;
- Анализ существующих мобильных приложений;
- Проектирование архитектуры приложения и базы данных;
- Реализация основных функциональных модулей приложения;
- Рассмотрение способов обеспечения безопасности системы;
- Оценка эффективности приложения на практике.

Методологическая основа исследования — труды отечественных и зарубежных учёных в области финансового менеджмента, проектирования информационных систем и разработки программного обеспечения. Также были изучены практические наработки по созданию мобильных приложений для управления личными финансами с учётом лучших практик.

Практическая ценность работы заключается в том, что разработанное мобильное приложение автоматизирует процесс управления и планирования

финансовых данных частного лица, способствует повышению уровня финансовой грамотности и позволяет эффективно организовать личный бюджет. Кроме того, приложение предоставляет пользователю возможность отслеживать доходы, расходы, накопления и финансовые цели, создавая тем самым благоприятные условия для анализа финансового состояния. Это помогает формировать личную финансовую дисциплину и в будущем позволяет снизить финансовые риски.

1 Теоретические основы учета и контроля личных финансов

1.1 Понятие и значение управления личными финансами

Управление личными финансами представляет собой комплексную деятельность, направленную на организацию, планирование и контроль собственных денежных средств. Этот процесс важен как для людей, располагающих избыточными финансовыми ресурсами, так и для тех, кому необходимо строго соблюдать бюджет из-за ограниченности доходов.

Как видно на схеме (рисунок 1), ключевые компоненты личных финансов включают в себя следующие элементы: получение дохода, расходы, накопления, вложения и страховую защиту [1].



Рисунок 1 – Сферы личных финансов

Доход — это поступление денежных средств, которое человек получает и использует для удовлетворения своих личных и семейных потребностей. Этот элемент считается отправной точкой в процессе финансового планирования, так как именно от него зависит дальнейшее распределение ресурсов. Основными источниками дохода могут выступать: заработная плата, пенсионные выплаты, дивиденды и прочие виды поступлений. Полученные финансы могут быть направлены на расходы, накопления или инвестиции.

Расходы охватывают все траты, связанные с приобретением товаров и услуг, необходимых для повседневной жизни. Это те средства, которые не идут

на создание капитала, а используются на потребление: питание, оплата налогов, кредитные обязательства, аренда жилья и прочее.

Сбережения, часто называемые «финансовой подушкой безопасности», представляют собой резерв, предназначенный на случай непредвиденных обстоятельств. Это может быть как наличность, так и средства на сберегательном счёте. Однако чрезмерное накопление денег без их вовлечения в оборот не считается эффективным, так как деньги должны приносить доход, а не просто «лежать без дела».

Инвестирование представляет собой процесс приобретения активов, от которых ожидается получение дохода в будущем. Цель инвестиций — приумножение первоначального капитала. Однако любые вложения связаны с определённой долей риска, поскольку не каждый актив гарантирует положительную доходность и прирост средств.

Страхование также относится к важным элементам управления личными финансами. Этот инструмент используется для защиты от возможных непредвиденных событий, которые могут нанести финансовый ущерб. Таким образом, страхование позволяет снизить уровень финансовых рисков в долгосрочной перспективе [2].

Первым и важнейшим этапом в процессе управления финансами является постановка целей. Необходимо чётко определить, каких финансовых результатов вы хотите достичь, и расставить приоритеты. Только после этого становится очевидным, насколько сложен и продолжителен путь к достижению поставленных задач. Целесообразно разделять долгосрочные и краткосрочные цели, чтобы иметь более структурированный подход к их реализации. Следующий шаг — оценка текущего финансового состояния. Для этого требуется проанализировать уровень доходов, регулярные расходы, объем сбережений и наличие задолженностей. Также необходимо составить перечень всех активов, чтобы понять, насколько они соответствуют установленным целям. Процесс управления личными средствами может быть формализован через финансовый план или бюджет. Одним из наиболее популярных подходов является метод 50/30/20, предполагающий распределение чистого дохода (после уплаты налогов) следующим образом:

- 50% — на базовые потребности: аренда жилья, коммунальные платежи, продукты, транспорт и другие жизненно важные расходы;
- 30% — на текущие желания и личные удовольствия: покупки, подарки, развлечения;
- 20% — на создание финансовой подушки, сбережения, инвестиции, пенсионные накопления и крупные цели [3].

По мере достижения ближайших (краткосрочных) целей приоритет постепенно смещается в сторону долгосрочных финансовых задач, таких как покупка недвижимости, обеспечение пенсии или создание пассивного дохода. Необходимо также учитывать оценку возможных альтернатив и связанных с ними рисков. Каждое финансовое решение влечёт за собой определённые

последствия и может исключать другие варианты действий. Например, вложение средств в акции может означать отказ от отпуска или другой запланированной траты. Чтобы грамотно учитывать риски, важно собирать информацию не только на основе собственного опыта, но и с опорой на практику других людей, а также использовать достоверные источники в области финансового планирования.

Для успешной реализации финансового плана может потребоваться поддержка со стороны профессионалов. Так, например, можно обратиться к страховому агенту для оформления страховки на имущество или к инвестиционному консультанту — для приобретения ценных бумаг. Сегодня также доступен широкий выбор цифровых инструментов для управления личным бюджетом. Практически каждый банк предлагает мобильные приложения и онлайн-платформы, с помощью которых можно настроить автоматическое перечисление части дохода на сберегательный счёт. Единственным условием является необходимость научиться распределять оставшиеся 80% бюджета таким образом, чтобы они покрывали все остальные расходы.

Важно понимать, что финансовое планирование — это не разовое действие, а непрерывный и гибкий процесс. Оно требует регулярного анализа и корректировки решений, особенно при изменении жизненных обстоятельств, экономических условий или социальных факторов. Такая адаптация позволяет оставаться в рамках намеченного курса и своевременно реагировать на возникающие вызовы.

Вопрос эффективного управления личными финансами неразрывно связан с необходимостью обеспечения безопасности на всех этапах — от получения дохода до его расходования и накопления. В связи с этим возникает закономерный вопрос: почему речь идет именно о личных финансах, а не, скажем, о семейных?

Первое, что стоит отметить — большинство современных финансовых и банковских инструментов ориентированы на индивидуального пользователя. Возможности коллективного или доверительного управления чаще всего предлагаются лишь как дополнительные функции. Во-вторых, защита личных финансов тесно связана с психологическими аспектами. Чтобы эффективно управлять общими ресурсами, человек должен сначала обеспечить устойчивость собственных активов — по принципу «сначала кислородную маску — себе, потом ребёнку».

В условиях современной экономики управление личными финансами приобретает особую актуальность, чему способствуют объективные обстоятельства, такие как:

- серьёзное влияние внешнеэкономических факторов (санкции, глобальные эпидемии) на уровень доходов;
- нестабильность цен на товары и услуги, сопровождающаяся трудностями долгосрочного прогнозирования;
- устойчивая тенденция к неравномерному перераспределению доходов — от активного накопления до чрезмерного потребления.

Кроме того, цифровизация экономики влечёт за собой новые вызовы. Среди них:

- нехватка цифровой грамотности у граждан, особенно в вопросах распоряжения личными средствами;
- рост угроз со стороны кибермошенников, что делает уязвимыми как текущие, так и долгосрочные финансовые активы.

В такой ситуации особенно важно формирование принципов так называемой «финансовой гигиены» — набора профилактических правил, позволяющих заранее снизить возможные риски и повысить устойчивость личного бюджета в долгосрочной перспективе.

Вопросы, связанные с управлением личными финансовыми ресурсами, активно изучаются как российскими, так и зарубежными исследователями. Так, в трудах С.А. Белозерова [1; 2], Р.С. Екшембиева [3], В.В. Глухова [4], А.А. Землянского и С.А. Бондаревой [5] акцент делается на значимости домашних хозяйств как самостоятельных субъектов в национальной финансовой системе, несущих полную ответственность за принимаемые финансовые решения.

Некоторые исследователи, в частности, Р.С. Екшембиев, А.А. Землянский и С.А. Бондарева, подчеркивают тесную связь между финансовым поведением индивида и его жизненным циклом, отмечая, что структура потребления и накоплений напрямую зависит от стадии жизненного пути.

Проблематика формирования доходов и их рационального использования рассматривается в научных трудах Л.М. Макаревича [6], Н. Фергюсона [7], а также G. Muske и M. Winter [8]. Вопросы планирования бюджета и постановки персональных финансовых целей раскрываются в исследованиях Ю.В. Сахаровской [9] и K.L. Hamilton [10].

Одним из ключевых факторов, влияющих на эффективность управления личными финансами, выступает уровень финансовой осведомлённости. В этом контексте представляет интерес работа L. Mandell и L.S. Klein [11], посвящённая оценке эффективности образовательных программ по финансовой грамотности. Особое значение в современных экономических условиях приобретает изучение инвестиционного поведения и восприятия риска со стороны физических лиц. Этими вопросами занимаются такие авторы, как А.О. Блинов [12], Ю.Ю. Финогенова [13], В.А. Кашин, В.В. Панков и В.И. Перов [14].

Финансовые решения, как правило, принимаются в условиях ограниченной информации и неопределённости, что зачастую приводит к иррациональному поведению. Люди склонны повторять одни и те же ошибки в мышлении и выборе. Этими феноменами занимаются как российские, так и зарубежные учёные. Среди них — лауреаты Нобелевской премии Д. Канеман [15] и Р. Талер [16], а также К. Санстейн [17], С.М. Reinhart и К.S. Rogoff [18].

1.2 Современные подходы к финансовому планированию

Управление личными финансами — это комплекс, состоящий из взаимосвязанных процессов, в которых человек, опираясь на уровень своего дохода, определяет направления расходования средств, а также виды и объем накоплений. Основная цель здесь — увеличение стоимости накопленного капитала. В этом процессе можно использовать современные технологии, однако в научных исследованиях они по-прежнему недостаточно рассмотрены.

В связи с этим далее рассмотрим некоторые методы управления личными финансами, особенно в контексте применения современных технологий, а также опишем подходы и методы личного финансового планирования [3].

Таблица 1

Схема управления личными финансами

Удовлетворение личных потребностей.	
Направления увеличения доходов: Заработная плата Пенсии и стипендии Доходы от сбережений и инвестиций Доходы от сдачи в аренду недвижимого и движимого имущества Выплаты и льготы от общественных организаций Доходы от личного подсобного хозяйства Доходы от предпринимательской деятельности и самозанятости Гранты, предоставляемые государственными и коммерческими структурами Подарки, выигрыши Доходы из других источников	Направления сокращения расходов: Продукты питания и безалкогольные напитки Алкогольные напитки и табачные изделия Одежда и обувь Жилищные услуги, вода, электроэнергия, газ и другие виды топлива Предметы для домашнего хозяйства, бытовая техника и средства повседневного ухода Здравоохранение Транспорт Связь Организация досуга и культурных мероприятий Образование Гостиницы, кафе и рестораны Прочие товары и услуги
Мобильные или стационарные приложения для учета личных финансов	
Учет, планирование и контроль личных финансов	
Защита личных финансов	

Неотъемлемой частью работы с личными финансами является процесс ведения учета, целевого планирования и контроля. Это динамичный процесс, требующий постоянного пересмотра и оценки. Он начинается с первоначальной оценки финансового положения, включает постановку целей и составление планов, а затем продолжается сохранением этих планов, мониторингом исполнения и повторной оценкой состояния после внесения изменений.

Подробное отслеживание личных финансов и регулярное обновление всех данных о доходах – это сложная и многоэтапная задача, с которой многим людям справиться непросто. Поэтому для повышения эффективности финансового управления целесообразно использовать специализированные финансовые приложения – как мобильные, так и стационарные. Сегодня на рынке представлено множество приложений от различных разработчиков, обладающих различным функционалом и уровнями аналитики. Главная цель таких приложений – упростить процесс управления личными финансами и повысить его эффективность. Эти приложения могут предлагать базовые функции, такие как отслеживание расходов и составление бюджета, что позволяет пользователю видеть, сколько и на что тратится. Кроме того, они оснащены дополнительными возможностями, например, автоматическим анализом банковских операций и предоставлением услуг финансового коучинга. Для этого такие приложения работают с многочисленными источниками данных и используют сложные алгоритмы. В результате пользователи могут автоматически отслеживать свои доходы и расходы, а также получать обзор своего финансового состояния. Помимо этого, они помогают ставить краткосрочные и долгосрочные финансовые цели и достигать их удобным способом. Всё это способствует повышению финансовой грамотности человека и формированию положительных финансовых привычек.

В Казахстане наблюдается активный рост использования приложений для управления личными финансами. Например, суперприложение **Kaspi.kz** является одним из самых популярных финансовых инструментов в стране. В 2021 году этим приложением пользовались более 7 миллионов казахстанцев, а треть транспортных средств в стране была зарегистрирована через эту платформу. В целом около 70% населения Казахстана (из примерно 20 миллионов человек) используют сервисы Kaspi.kz.

По данным исследований 2021 года, уровень финансовой грамотности среди взрослого населения Казахстана составил 59,6%. Это один из самых высоких показателей среди стран СНГ. Кроме того, Казахстанская фондовая биржа (KASE) реализует различные образовательные программы по повышению финансовой грамотности. В 2021 году в конкурсе «Симулятор фондового рынка» приняли участие около 2 000 студентов из более чем 60 университетов. Эти данные свидетельствуют о положительных изменениях в сфере финансовых технологий и образования в Казахстане. Однако для дальнейшего повышения финансовой грамотности и более широкого применения инструментов управления личными финансами необходимы дополнительные меры.

Еще один эффективный способ повышения личной эффективности граждан в сфере личных финансов — сокращение неэффективных расходов. Это одна из самых актуальных проблем в управлении личными финансами. Однако, вопреки первоначальному впечатлению, решение этой проблемы не столь сложное. Основной путь сокращения неэффективных расходов — это не резкое снижение затрат, а постепенное уменьшение по каждой категории. Здесь может помочь привычка осознанных покупок, отказ от импульсивных трат, а также ведение учета расходов по разным категориям.

Еще одним важным инструментом эффективного управления личными финансами является составление бюджета. Создание и соблюдение бюджета считается одним из ключевых навыков в управлении личными средствами, однако на практике лишь около трети людей составляют подробный бюджет. Одним из самых популярных и эффективных методов является «правило 50 – 30 – 20». Суть этого правила следующая:

- 50% ежемесячного дохода должно идти на основные жизненные потребности — аренду жилья или ипотеку, коммунальные услуги, продукты питания и т.п.;
- 20% направляется на сбережения, создание резервного фонда и/или инвестиции;
- 30% выделяется на желательные, но необязательные траты — например, одежду, билеты на концерты, посещение ресторанов или кафе [6].

Это правило помогает сбалансированно и эффективно управлять финансами, обеспечивая финансовую стабильность личности. Существуют различные вариации данного процентного распределения, и каждый человек может адаптировать его в соответствии со своими доходами и финансовыми предпочтениями.

Одним из подходов, заслуживающих особого внимания в управлении личными финансами, является регулярное инвестирование. Мониторинги и отчеты показывают, что финансово успешные люди не ограничиваются только сокращением расходов и накоплением средств. Они переходят на следующий уровень и направляют сэкономленные средства на инвестиции. Такая модель инвестиционного мышления позволяет регулярно увеличивать часть накопленного капитала, что в свою очередь может значительно повысить благосостояние человека. Регулярные накопления со временем, даже в небольших объемах, могут привести к ощутимым результатам. Однако постоянное инвестирование требует финансовой дисциплины — необходимо быть готовым вкладывать средства не в текущие потребности, а в будущее. Опытные инвесторы умеют оценивать, какие инвестиции целесообразны именно сейчас, а какие стоит отложить, и управляют своим инвестиционным портфелем, исходя из экономических тенденций и результатов предыдущих вложений.

Для людей, не имеющих достаточного опыта в управлении личными финансами или только начинающих распоряжаться своими финансовыми ресурсами, разумно обратиться за помощью к финансовому консультанту. Финансовый консультант может предоставить знания и информацию, необходимые для составления эффективного личного финансового плана.

Однако перед тем как выбрать консультанта, важно уточнить содержание его услуг и структуру оплаты, а также собрать отзывы и информацию о его профессиональном опыте. Это является основным условием получения качественных и надежных услуг.

Одним из ключевых аспектов управления личными финансами является обеспечение финансовой безопасности. В большинстве случаев граждане уделяют недостаточное внимание безопасности личных финансов, особенно в цифровой среде. Это может привести к беззаботному отношению к мошенничеству и информационным угрозам. В связи с этим исследователи, такие как Л. А. Родина и Л. В. Завьялова, подчеркивают актуальность усиления мер безопасности в этой сфере. В своих работах Завьялова отмечает, что в условиях цифровизации «правила личной финансовой гигиены» представляют собой совокупность простых мер предосторожности, направленных на защиту данных владельцев финансовой информации при использовании цифровых технологий в сфере личных финансов.

Эти правила не рассматриваются как полное и единственное решение, поскольку в современном мире цифровизация активно проникает во все сферы, а инструменты и технологии управления личными финансами постоянно обновляются. В первую очередь необходимо четко определить объект и субъект применения правил финансовой гигиены. По мнению исследователей, эти правила следует ориентировать на банковские карты, мобильный банкинг и движение финансовой информации [7].

В заключение можно отметить, что для повышения эффективности управления личными финансами необходимо применять комплексный подход. Этот подход должен включать современные технологии, методы планирования и прогнозирования, а также повышение финансовой грамотности. Такой подход позволяет человеку принимать осознанные финансовые решения, оценивать инвестиционные риски и минимизировать их, а также обеспечивать личную финансовую безопасность.

1.3 Анализ существующих мобильных приложений для учета финансов

В эпоху цифровизации управление личными финансами стало насущной необходимостью для широких слоев населения. Сложные экономические условия и рост безналичных расчетов стимулируют людей более осознанно подходить к планированию бюджета. Поэтому мобильные приложения для учета доходов и расходов приобрели большую популярность во всем мире. Как отмечают исследователи, управление личными финансами — весьма актуальная тема на сегодняшний день, и интерес к ней постоянно растёт. По данным опросов, до 80% пользователей смартфонов регулярно пользуются различными финансовыми приложениями, а примерно треть из них применяет одновременно

три и более таких сервисов. В Казахстане также наблюдается высокий уровень внедрения финансовых технологий: например, самое популярное в стране суперприложение Kaspi.kz ежемесячно насчитывает свыше 11 млн активных пользователей, что составляет около 60% населения Казахстана. Эти факты подтверждают востребованность мобильных решений для управления личными финансами и их значимость для повышения финансовой грамотности населения.

Современные мобильные приложения для учета личных финансов предлагают пользователям широкий спектр возможностей. К основным функциям таких приложений относятся:

- Автоматизированный учет транзакций и расходов. Приложения могут подключаться к банковским счетам пользователя и автоматически импортировать данные о тратах и поступлениях, избавляя от необходимости ручного ввода. К примеру, приложение Spendee позволяет связать банковские счета и автоматически загружает все транзакции, предоставляя единый обзор денежных потоков. Многие приложения поддерживают импорт данных из электронных кошельков и даже криптовалютных аккаунтов.

- Бюджетирование и планирование расходов. Пользователю предоставляются инструменты для составления бюджета на месяц или другой период, установки лимитов по категориям трат и отслеживания их выполнения. Например, приложение Goodbudget реализует метод «конвертов», когда пользователь заранее распределяет сумму по категориям бюджета и затем контролирует траты в рамках каждого конверта. Другие сервисы, такие как NerdWallet, предлагают рекомендации по распределению бюджета (например, по правилу 50/30/20) и помогают оценивать баланс между расходами и сбережениями.

- Анализ данных и визуализация в графиках. Практически все популярные приложения оснащены средствами визуализации финансовой информации – диаграммами, графиками трендов, «пирогам» по категориям расходов. Это позволяет пользователю наглядно увидеть структуру своих трат, долю каждой категории (питание, транспорт, коммунальные услуги и т.д.) и динамику накоплений или долгов. Благодаря этому финансовые привычки становятся более понятными для пользователя, что подтверждается отзывами: красивые и простые графики делают финансовую картину легко воспринимаемой.

- Мультивалютность и конвертация валют. В условиях глобализации ряд приложений поддерживает работу с разными валютами. Это важно для пользователей, получающих доход или совершающих расходы в иностранной валюте, а также для путешественников. Например, приложение Buxfer позволяет учитывать транзакции в разных валютах и автоматически пересчитывать их, обеспечивая актуальный баланс.

- Напоминания и планирование платежей. Многие приложения предлагают функцию установки напоминаний о предстоящих платежах (кредиты, счета за услуги) или контроль сроков оплаты. Так, EveryDollar и ряд других программ умеют напоминать о приближении даты платежа по счету либо предупреждать, когда бюджет по категории близок к превышению лимита.

- Дополнительные «фишки» и интеграции. Разработчики внедряют и более инновационные функции. Некоторые приложения позволяют сканировать чеки и распознавать данные расходов по фотографиям, облегчая ввод информации. Другие, например Budget Planner, реализуют распознавание данных о расходах с голоса, когда пользователь может продиктовать сумму и категорию траты. Отдельные сервисы дают прогнозы финансового состояния – на основе истории транзакций формируется прогноз денежных остатков на несколько месяцев вперед, что помогает планировать крупные покупки. Также заслуживает внимания функция совместного ведения бюджета: Spendee и некоторые аналогичные приложения позволяют создавать общие кошельки для семьи или группы людей, чтобы коллективно планировать бюджет и отслеживать расходы.

Преимущества. У использования мобильных финансовых приложений есть ряд очевидных преимуществ:

- *Удобство и оперативность.* Смартфон всегда под рукой, поэтому в любой момент можно проверить состояние бюджета, ввести новую трату или скорректировать планы. Приложения дают моментальный доступ к финансовой информации, тогда как традиционные методы учета (например, Excel) требовали больше времени и не были столь мобильными.

- *Повышение финансовой дисциплины.* Постоянный контроль расходов и наличие наглядной статистики помогают выработать полезные привычки – меньше тратить на импульсивные покупки, откладывать сбережения. Исследования показывают, что пользователи, активно пользующиеся бюджетными приложениями, получают более глубокое понимание своих финансов и чаще добиваются поставленных финансовых целей.

- *Персональные советы и образовательный эффект.* Некоторые приложения (например, NerdWallet, Mint) помимо учета данных дают пользователю рекомендации: как оптимизировать траты, на чем можно сэкономить, какие финансовые продукты (вклады, кредиты) подходят под его профиль. Это повышает финансовую грамотность. А доступ к архиву данных (некоторые сервисы хранят историю за несколько лет) позволяет проанализировать свои финансовые тенденции во времени и сделать выводы для будущего.

- *Безопасность и снижение риска ошибок.* Ведя учет вручную, человек может допустить ошибки в расчетах. Приложения же автоматически суммируют и классифицируют транзакции, уменьшая вероятность ошибок. Кроме того, использование безналичных методов через приложения снижает риск потери наличных или кражи кошелька; цифровые финансы более защищены от бытовых рисков, хотя и требуют внимания к кибербезопасности.

Недостатки. Вместе с тем, существуют и недостатки/риски использования таких приложений:

- *Конфиденциальность и безопасность данных.* Для полноценной работы многие приложения запрашивают доступ к банковским счетам или хранят на своих серверах подробные сведения о финансовой жизни

пользователя. Это рождает риски утечки данных или несанкционированного доступа. Пользователь фактически доверяет свои финансовые данные стороннему сервису, что требует уверенности в надежности разработчиков и соблюдении ими мер защиты. Эксперты отмечают, что низкий уровень киберграмотности и излишняя уверенность пользователей могут сыграть злую шутку, если пренебречь правилами безопасности.

- *Зависимость от технологий.* Если приложение работает только при наличии интернета или на определенном устройстве, пользователь становится зависим от технических условий. Сбой сервера, потеря смартфона или просто отсутствие сети могут временно лишить доступа к бюджету. Это ставит под угрозу непрерывность учета. Также некоторые пользователи отмечают, что из-за автоматизации они сами хуже запоминают свои финансовые операции, полностью полагаясь на приложение.

- *Необходимость дисциплины.* Парадоксально, но для эффективного использования бюджета через приложение тоже нужна самодисциплина. Нередко пользователи начинают с энтузиазмом вести учет, но затем забрасывают приложение, если оно требует ручного ввода каждой траты. Без регулярного внесения данных статистика теряет актуальность. Согласно исследованиям, многие перестают активно пользоваться бюджетным приложением спустя несколько месяцев, если не видят мгновенного результата или испытывают сложности в освоении интерфейса.

- *Ограниченная локализация и функциональность.* Международные приложения не всегда учитывают специфику локальных финансовых систем. Например, в некоторых странах они не могут подключиться к местным банкам, не поддерживают национальную валюту или язык интерфейса. В результате в определенных регионах их полезность снижается. Так, по итогам обзора существующих решений, проведенного Стефановым и др., выяснилось, что лишь немногие приложения удовлетворяют потребности пользователей в простом управлении финансами, и практически отсутствуют варианты на национальном (локальном) рынке. Это стимулирует разработчиков создавать локальные продукты, адаптированные под требования конкретной страны.

В настоящее время существует множество приложений для учета личных финансов, каждое со своим набором функций и особенностей. В качестве примера в таблице 1 приведено сравнение ключевых характеристик девяти популярных международных приложений, проанализированных в недавнем исследовании (2024). Среди них – как широко известные глобальные сервисы (Mint, Spendee, Goodbudget и др.), так и специализированные решения для бюджетирования и оплаты счетов.

Таблица 2 – Сравнение функций популярных приложений для учета личных финансов

Приложение	Ключевые функции	Модель монетизации	Сложность освоения	Синхронизация с банками
Mint	Отслеживание наличных расходов и баланса счетов; бюджеты; отчеты по транзакциям; налоговая информация; поддержка криптовалют и инвестиций	Условно-бесплатное (база бесплатна, премиум-опции)	Средняя	Да
Buxfer	Ручной ввод транзакций; установка бюджетов; контроль расходов; поддержка разных валют; уведомления о счетах и превышении лимитов	Условно-бесплатное (бесплатная базовая версия, расширенные платные планы)	Низкая (легко)	Да
Goodbudget	Метод «конвертов» (предварительное распределение средств по категориям); ручной учет транзакций; корректировка бюджета; перенос остатка на следующий период; хранение данных за 7 лет	Условно-бесплатное (бесплатная версия, есть платный план расширения)	Средняя	Да

NerdWallet	Импорт транзакций с банковских и инвестиционных счетов; ручное планирование бюджета (50/30/20); советы по экономии и оптимизации расходов; мониторинг кредитных карт	Бесплатное	Средняя	Да
EveryDollar	Планирование и корректировка бюджета на период; ручное добавление расходов; создание фондов сбережений; напоминания о платежных сроках	Платная подписка (~\$79.99/год или \$12.99/мес, есть пробный период)	Низкая (легко)	Да
Mobills	Учет наличных расходов и лимитов по картам; графическая статистика расходов; напоминания о финансовых дедлайнах; отображение доступного остатка	Условно-бесплатное (бесплатно с опциональным и премиум-обновлениями)	Низкая (легко)	Да
Spendee	Ведение личных финансовых счетов; совместные бюджеты с	Условно-бесплатное (базовые функции бесплатны,	Низкая (легко)	Да

	другими пользователями; поддержка разных валют; повышенная защита данных; веб-версия для ПК	премиум-подписка)		
Budget Planner (<i>Expense Tracker</i>)	Планирование месячного бюджета; хранение данных о доходах/расходах ; ввод данных голосом; автоплатежи (повторяющиеся транзакции по расписанию); финансовый прогноз на 6 месяцев; диаграммы ежедневных расходов	Условно-бесплатное (бесплатно, есть премиум-функции)	Средняя	Да
Prism (<i>Pay Bills</i>)	Автоматическое отслеживание счетов и сроков оплат; история финансовых транзакций; календарь платежей; напоминания о платежах; защита данных (шифрование, PIN/биометрия)	Бесплатное	Средняя	Да

Примечание: Все перечисленные приложения доступны на популярных мобильных платформах (Android, iOS). Условно-бесплатная модель означает наличие бесплатного базового функционала с возможностью приобретения

расширенных функций (Premium) за дополнительную плату. Сложность освоения оценивалась экспертами: «низкая» соответствует интуитивно понятному, простому в использовании интерфейсу, «средняя» – наличию некоторой сложности или обилия функций, требующих времени на освоение пользователем.

Из таблицы видно, что наиболее распространённые приложения предлагают сходный базовый набор функций (учет доходов и расходов, бюджеты, синхронизация с банковскими счетами) при некоторых отличиях в специализации. Например, Goodbudget ориентирован на методику конвертного бюджетирования, EveryDollar – на философию нулевого бюджета и систему финансов Дейва Рамси, Prism фокусируется на оплате счетов и календарном планировании обязательных платежей. По модели распространения преобладает freemium (условно-бесплатная) схема: пользователю доступны бесплатные возможности, достаточные для базового использования, а за расширенные функции (синхронизация более чем с двумя счетами, совместная работа, дополнительные аналитические отчеты и т.п.) взимается плата по подписке. Лишь EveryDollar практически полностью коммерческий (расширенная версия требует оплачиваемой подписки).

Отдельно стоит подчеркнуть реализацию инновационных функций в рассмотренных приложениях. Графическая визуализация данных (диаграммы, отчёты) присутствует практически во всех решениях, что стало стандартом де-факто для подобных сервисов. Конвертер валют и поддержка мультивалютных счетов имеются в приложениях, рассчитанных на международное использование (Spendee, Buxfer и др.), что важно для пользователей с транзакциями в нескольких валютах. Сканирование чеков и автоматическое распознавание данных по фото внедрено не во всех приложениях – такая функция чаще встречается в специализированных менеджерах расходов (например, Expensify, Wave) или корпоративных решениях, однако на массовом рынке ее начинают внедрять для удобства пользователей, чтобы сократить ручной ввод данных. Кроме того, появляющиеся интеграции с голосовыми помощниками и ИИ также можно считать новым трендом: например, Halyk Bank анонсировал функцию голосового управления платежами в своем приложении и одновременное добавление модуля «персонального финансового планирования», позволяющего каждому видеть аналитику своих расходов. Такие технологии, как голосовой ввод данных о затратах (реализованный в Budget Planner) и интеллектуальные подсказки, повышают удобство и делают взаимодействие с финансовым приложением еще более простым и естественным.

Казахстанские решения и локальный контекст

Мировые практики находят отражение и в Казахстане: местные банки и финтех-компании внедряют функциональность учета личных финансов в свои продукты. В экосистеме Halyk Bank (одного из крупнейших банков РК) ведется работа над инструментами финансовой аналитики для клиентов – планируется, что мобильное приложение банка будет автоматически анализировать расходы пользователя по категориям и предоставлять ему персональные инсайты для

улучшения управления бюджетом. Jusan Bank в обновленной версии своего приложения реализовал возможность совместного бюджета: клиенты могут открывать совместные счета для семьи с общим доступом и лимитами, что облегчает ведение семейного бюджета. Также Jusan предлагает объединенную историю транзакций по всем продуктам клиента, позволяя в одном месте отслеживать расходы по картам, счетам, кредитам и т.д., что повышает прозрачность личных финансов. Freedom Bank, развивающий свою экосистему *Freedom SuperApp*, фокусируется преимущественно на инвестициях и банковских операциях, однако и там появляются элементы управления расходами (например, мгновенные уведомления о тратах, интегрированный конвертер валют, анализ покупок по категориям в выписке).

Отдельно от банковского сектора существуют независимые приложения на русском языке, популярные среди казахстанских пользователей. Это, в частности, такие сервисы как Monefy, Дзен-мани (Zenmoney), CoinKeeper и др., изначально разработанные в соседних странах (Россия, Украина) и адаптированные под СНГ. Они позволяют вести учет наличных и безналичных расходов, часто умеют считывать SMS-уведомления от банков для автоподстановки операций (что актуально для локальных банков, не поддерживающих прямую интеграцию через API). Однако массовый пользователь в Казахстане пока больше склоняется к использованию встроенных возможностей приложений своих банков. Супераппы вроде Kaspi.kz фактически стали универсальным финансовым инструментом, объединяющим банковские услуги, платежи, покупки и базовый контроль баланса. Тем не менее, специализированные приложения для детального бюджетирования занимают свою нишу среди более продвинутых и заинтересованных в планировании аудитории.

При сравнении казахстанских приложений с международными аналогами можно отметить, что функционально разрывы сокращаются. Если еще несколько лет назад локальные банковские приложения предоставляли лишь выписку по счету, то сегодня они начинают предлагать категории расходов, графики, советы. Международные же приложения продолжают задавать тон в отдельных возможностях (например, интеграции с большим числом внешних сервисов, продвинутый анализ данных). Важно, что локальные решения учитывают особенности языка, валюты и банковской инфраструктуры, что дает им преимущество для рядового пользователя в Казахстане.

Анализ существующих мобильных приложений для управления личными финансами показывает, что они стали неотъемлемым инструментом для многих людей по всему миру. Высокая востребованность этих сервисов обусловлена их очевидной пользой: они повышают финансовую дисциплину, экономят время и упрощают процесс бюджетирования. Популярные международные приложения (Mint, Goodbudget, Spendee, и др.) предлагают богатый функционал: от синхронизации с банковскими счетами до тонкой настройки бюджета и наглядных отчетов. Их опыт показывает, какие функции наиболее ценны для

пользователей – и эти же подходы постепенно перенимают локальные финтех-решения.

Перспективы развития приложений для учета личных финансов остаются очень высокими. Рынок продолжает расти двузначными темпами, а сами приложения эволюционируют, внедряя инновации вроде искусственного интеллекта, прогнозной аналитики и бесшовной интеграции с экосистемами банков. В Казахстане и других странах появляются собственные продукты или модули в банкинге, ориентированные на финансовое планирование, что говорит о признании потенциала этих инструментов. Можно ожидать, что в ближайшие годы функциональность таких приложений станет еще шире, а спрос на удобные цифровые помощники по личным финансам будет только расти. Это позитивный тренд, поскольку массовое распространение подобных технологий способствует повышению финансовой грамотности и благосостояния населения в целом.

2 Проектирование мобильного приложения для контроля личных финансов

2.1 Определение функциональных требований

Одним из важнейших этапов эффективного проектирования информационной системы и мобильного приложения является чёткое и системное определение функциональных требований. На этом этапе уточняется структура будущей системы, логика её работы, а также функции, соответствующие потребностям пользователя.

В мобильном приложении для управления личными финансами эти требования должны охватывать повседневные действия, такие как фиксация доходов и расходов, контроль затрат, проведение анализа и финансовое планирование. В целях повышения финансовой грамотности необходимо предусмотреть функции распределения расходов по категориям, построения аналитических диаграмм, постановки целей и отправки напоминаний. Все эти возможности должны быть реализованы через удобный пользовательский интерфейс.

Точность функциональных требований позволяет адаптировать систему под целевую аудиторию, согласовать технические решения с интерфейсом UX/UI и избежать избыточного кода и дублирования функциональности.

Таким образом, корректное определение функциональных требований является основой успешной реализации проекта и формирования основных функциональных возможностей системы.

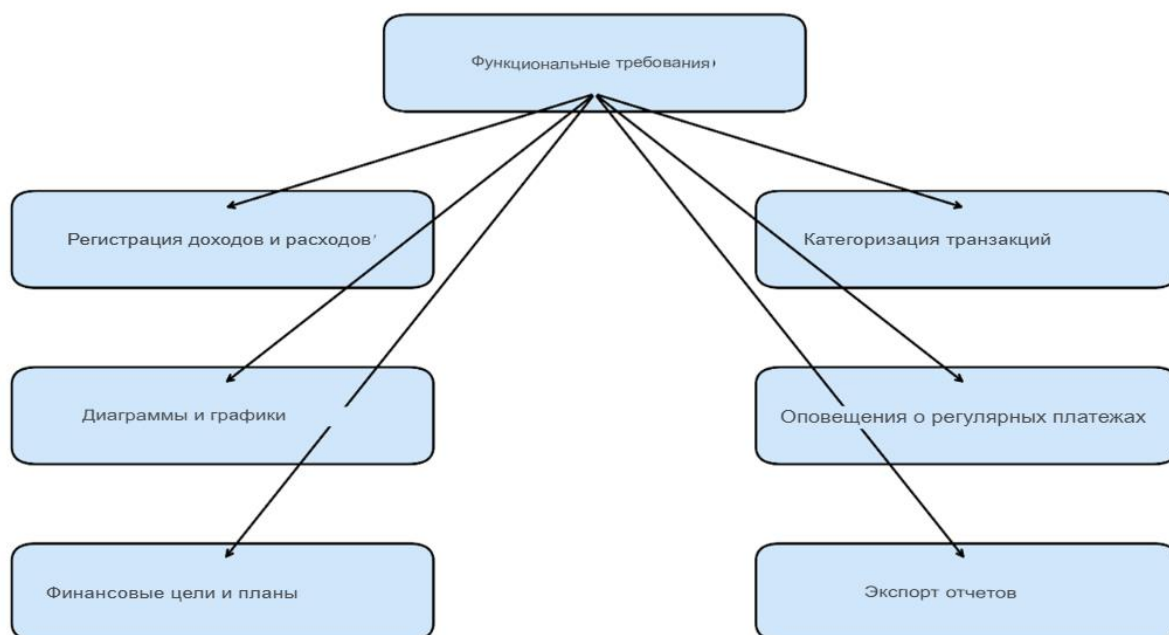


Рисунок 2 - Функциональные требования к приложению

Регистрация доходов и расходов

Основное условие соблюдения финансовой дисциплины и эффективного управления личным бюджетом — это точная и своевременная регистрация всех доходов и расходов. Эта функция является одним из ключевых компонентов мобильного приложения для управления личными финансами. Пользователь может ввести в систему любую финансовую операцию — как источники дохода (зарплата, стипендия, премия), так и расходы (еда, транспорт, коммунальные услуги), указывая точную сумму и дату.

Процесс регистрации операций осуществляется через простой и доступный интерфейс. Для каждой транзакции выбирается тип: «доход» или «расход». При необходимости операция может быть отнесена к определённой категории (например: «продукты», «образование», «транспорт»). Эти данные аккумулируются в системе и служат основой для последующего финансового анализа и отчётности.

Используя эту функцию, пользователь может видеть структуру своих расходов и определить, на какие категории уходит больше всего средств. Также появляется возможность оценить баланс доходов и расходов в ежемесячном или еженедельном разрезе.

Классификация транзакций по категориям

Один из эффективных методов управления личными финансами — это классификация всех финансовых операций по тематическим или целевым категориям. Эта функция позволяет присваивать каждой транзакции (доходу или расходу) определённую категорию, например: «продукты», «жильё», «транспорт», «развлечения», «зарплата» и т.д.

Классификация даёт пользователю возможность всесторонне анализировать структуру своих доходов и расходов. Например, определив, на какую категорию в определённый период ушло больше всего средств, можно скорректировать финансовую дисциплину. Сравнение расходов по категориям помогает выявить и сократить ненужные траты.

Функция тесно связана с аналитическими инструментами: по каждой категории строятся диаграммы и графики, что упрощает восприятие информации. Категории могут быть организованы иерархически (основные и подкатегории), что даёт возможность для более глубокого анализа.

Таким образом, классификация транзакций — это важный инструмент для оптимизации расходов, повышения финансовой дисциплины и планирования бюджета.

Построение диаграмм и графиков

Визуализация финансовой информации — это эффективный способ, способствующий быстрому и лёгкому пониманию динамики доходов и расходов, соблюдения бюджета и процесса накоплений. Поэтому в мобильном приложении предусмотрена функция автоматического построения диаграмм и графиков.

Эта функция на основе всех зарегистрированных транзакций предоставляет пользователю различные визуальные аналитические

инструменты. Например, круговая диаграмма показывает распределение расходов по категориям, а линейные или столбчатые графики демонстрируют финансовые изменения за определённый период — ежемесячный доход, расход и динамику баланса.

Визуальная аналитика помогает пользователю сосредоточиться не только на числовых значениях, но и на взаимосвязях между ними, что особенно полезно при принятии финансовых решений, оптимизации расходов или планировании целей накопления.

Диаграммы и графики обновляются в реальном времени и предоставляются в индивидуализированном формате, что создаёт благоприятную среду для контроля за финансовым поведением и долгосрочного планирования.

Таким образом, визуальное представление данных повышает функциональную ценность приложения и усиливает ответственность и контроль пользователя за своими финансами.

Напоминания о регулярных платежах

Одним из ключевых элементов эффективного управления личными финансами является своевременное выполнение регулярных платежей — таких как аренда жилья, коммунальные услуги, кредитные выплаты, подписки и т.п. Поэтому в приложении реализован механизм уведомлений и напоминаний, направленный на поддержание финансовой дисциплины пользователя и снижение риска просрочки платежей.

Пользователь может указать тип регулярного платежа и задать его повторение (ежедневно, еженедельно, ежемесячно и т.д.). На основе этой информации система автоматически отправляет push-уведомление или внутреннее напоминание о предстоящем платеже.

Время напоминания настраивается по желанию пользователя — например, за день до платежа или в день платежа. Это повышает финансовую ответственность и помогает избежать незапланированных расходов.

Кроме того, система может вести учёт невыполненных или отложенных платежей и предлагать пользователю рекомендации по улучшению финансовой дисциплины.

В целом, напоминания о регулярных платежах — это надёжный инструмент автоматизации финансового управления и обеспечения своевременных оплат, особенно важный для забывчивых пользователей или тех, у кого много финансовых обязательств.

Ввод финансовых целей и планов

В системе управления личными финансами важным шагом является постановка краткосрочных и долгосрочных финансовых целей пользователем. Это способствует повышению финансовой грамотности и формированию привычки к накоплениям. В связи с этим в приложении реализована функция ввода и отслеживания финансовых целей.

С её помощью пользователь может задать конкретную цель, например: поездка в отпуск, покупка бытовой техники, получение образования или

формирование резервного фонда. Для каждой цели указывается планируемая сумма, срок достижения и текущий объём накоплений. Система на этой основе отображает прогресс в процентах, наглядно показывая ход накоплений.

Пользователь может узнать, сколько необходимо откладывать регулярно, чтобы достичь цели в заданный срок. Это повышает мотивацию и способствует систематическому накоплению средств. Также приложение позволяет отслеживать историю и динамику изменений по каждой цели.

Финансовые цели визуализируются в виде графиков или шкал прогресса, что делает путь к цели понятным и наглядным. В некоторых случаях система может уведомить пользователя о задержке в достижении цели и предложить рекомендации.

Таким образом, функция постановки целей — это стратегический инструмент управления личными финансами и один из эффективных способов формирования финансовой дисциплины.

Возможность экспорта отчётов

Для эффективного контроля над финансовыми операциями и проведения долгосрочного анализа необходима возможность сохранения и обмена отчётами во внешних форматах. В мобильном приложении предусмотрена функция экспорта финансовых отчётов.

Пользователь может выгрузить сводные данные о доходах и расходах за определённый период в формате PDF, Excel или CSV. В отчётах указываются дата каждой транзакции, её категория, сумма и тип (доход/расход). Кроме того, к отчёту могут прилагаться сводные таблицы и графики.

Экспорт отчётов может быть полезен в следующих случаях:

- для подведения годового или месячного итога по личным финансам;
- для бухгалтерской или налоговой отчётности;
- для обмена информацией с финансовыми консультантами или членами семьи.

Эта функция особенно актуальна для пользователей, применяющих приложение в профессиональных или бизнес-целях. Также может быть предусмотрена возможность загрузки отчётов в облачные сервисы хранения (Google Drive, Dropbox и др.).

Таким образом, экспорт отчётов повышает гибкость приложения, позволяет пользователю полноценно работать с данными и принимать финансовые решения на основе достоверной информации.

Таблица 3

Структура взаимосвязи функциональных компонентов системы

№	Название функции	Связанные компоненты (подсистема / другие функции)	Роль пользователя	Результаты на выходе
1	Регистрация доходов и расходов	Пользовательский интерфейс, база транзакций, аналитика	Все пользователи	Зарегистрированные операции
2	Классификация транзакций по категориям	Форма ввода, модель данных, модуль диаграмм	Все пользователи	Классифицированные операции
3	Построение диаграмм и графиков	Данные транзакций, аналитическая система, временные фильтры	Все пользователи	Визуальные отчёты, графики
4	Напоминания о регулярных платежах	График платежей, система push-уведомлений	Все пользователи	Своевременные напоминания
5	Финансовые цели и планы	Модель целей, данные накоплений, индикатор прогресса	Все пользователи	Прогресс накоплений, мотивационные отчёты
6	Экспорт отчётов	База транзакций, аналитика, модуль экспорта (PDF/Excel)	Все пользователи	Отчёты в формате документов (PDF, Excel)

Данная таблица описывает основные функциональные модули мобильного приложения для управления личными финансами и их взаимосвязь. Каждая функция работает во взаимодействии с конкретными компонентами (подсистемами или другими функциями) и выполняет задачи в соответствии с ролью пользователя. Также указано, какие данные или действия являются результатом выполнения каждой функциональности.

В таблице охвачены ключевые функции, такие как регистрация доходов и расходов, классификация транзакций по категориям, построение графиков, напоминания о регулярных платежах, постановка финансовых целей и экспорт отчётов. Эта структура полноценно отражает логические и технологические

связи между модулями приложения и демонстрирует, что система является комплексной и удобной для пользователя.

В целом, определение функциональных требований является одним из основных этапов проектирования мобильного приложения и обеспечивает структурную целостность системы и её ориентацию на удобство пользователя. Основные предложенные функции приложения для управления личными финансами — регистрация доходов и расходов, классификация транзакций, построение аналитических диаграмм, напоминания о регулярных платежах, установка финансовых целей и экспорт отчётов — способствуют улучшению финансовой дисциплины пользователя и принятию осознанных решений.

2.2 Архитектура приложения

Система построена на основе клиент-серверной архитектуры. Такой подход широко используется при проектировании современных мобильных и веб-приложений и позволяет эффективно организовать структуру системы. В клиент-серверной модели вся бизнес-логика и обработка данных происходят на стороне сервера, а пользователь взаимодействует только с готовыми результатами через интерфейс. Этот метод обеспечивает масштабируемость, безопасность и модульную структуру системы.

Данная архитектура основана на разделении данных и функциональности на две части — клиентскую (сторона пользователя) и серверную (логика и хранение данных). Такая модель позволяет защищать данные, эффективно использовать сетевые ресурсы, обеспечивает работу на различных устройствах (кроссплатформенность), а также создаёт условия для отдельной разработки и обновления каждого компонента.

Архитектура приложения включает в себя следующие основные части:

- Фронтенд (пользовательский интерфейс);
- Бэкенд (серверная логика);
- База данных;
- Система безопасности.

Фронтенд (пользовательский интерфейс)

Пользовательский интерфейс — это основной элемент взаимодействия между пользователем и системой. В данном проекте фронтенд реализован с использованием шаблонной системы фреймворка Django. Эта система формирует веб-страницы с помощью HTML, CSS и JavaScript и динамически отображает данные, полученные с сервера.

Основная задача фронтенда — предоставить пользователю удобный способ работы с системой. Через интерфейс пользователь может зарегистрироваться, войти в систему, добавлять финансовые операции, отслеживать расходы, устанавливая цели и просматривать аналитические

данные. На каждой странице поддерживается простой и понятный макет: верхнее меню, поля для ввода данных, кнопки и таблицы с отображением результатов.

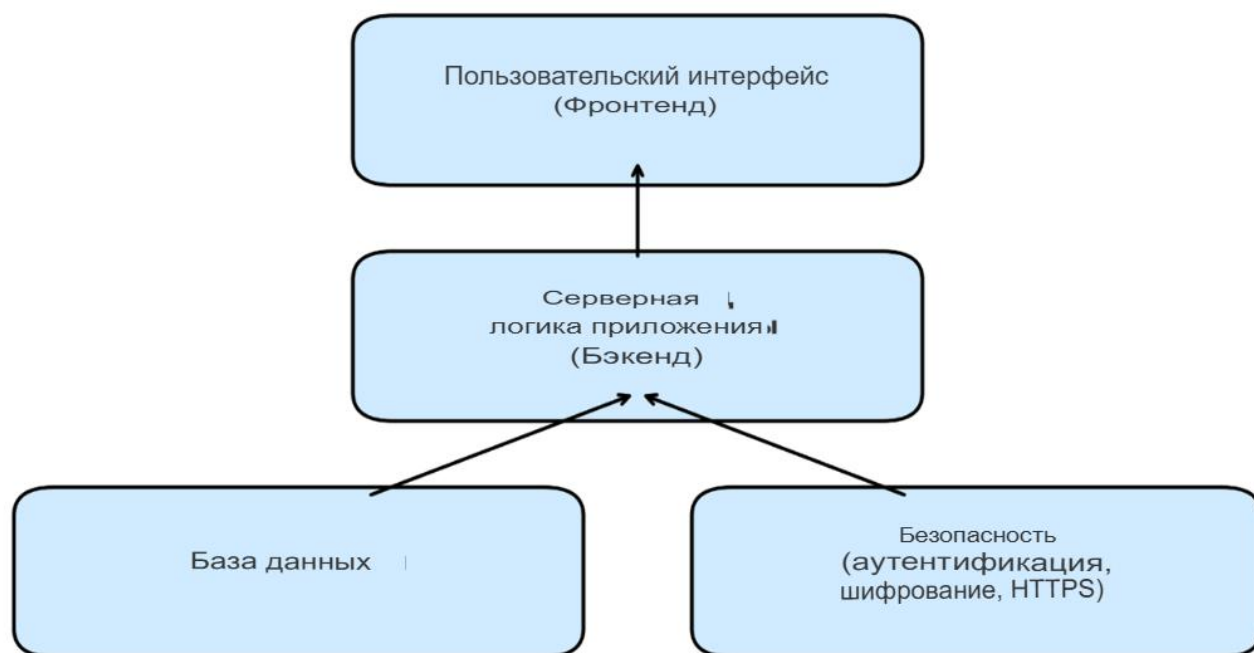


Рисунок 3 – Клиент-серверная архитектура приложения

Данное архитектурное решение обеспечивает надёжную работу системы, чёткую организацию логических связей между функциональными компонентами и высокий уровень пользовательского опыта.

В HTML-шаблонах данные отображаются с помощью функции `render()` из Django, связывающей шаблоны с серверной логикой. Внутри шаблонов используются циклы и условные операторы, что позволяет адаптировать отображаемое содержимое в зависимости от действий пользователя. Информация, введённая через формы, отправляется на сервер, где обрабатывается и возвращается обратно в виде ответа.

Также для стилизации интерфейса могут использоваться CSS-библиотеки, включая фреймворк Bootstrap. Это делает дизайн приложения согласованным, современным и адаптивным к мобильным устройствам.

Представленная ниже схема поэтапно демонстрирует взаимодействие пользователя с фронтенд-частью системы. Процесс начинается с того, что пользователь устанавливает соединение с системой через веб-интерфейс. Он просматривает страницы, основанные на HTML, вводит необходимую информацию в специальные формы или выполняет действия с помощью кнопок.

В результате этих действий запрос обрабатывается через шаблон Django и отправляется на сервер по протоколу HTTP. Серверная логика обрабатывает данные и возвращает результат. Возвращённый ответ формируется как веб-

страница и отображается пользователю в виде новых данных или сообщения. Этот процесс представлен в схеме в виде логического потока сверху вниз.

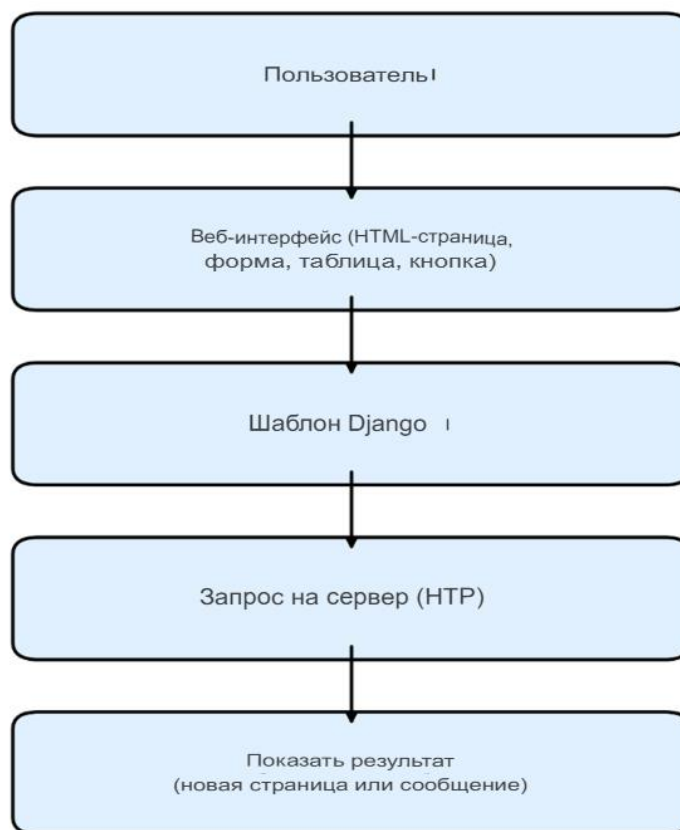


Рисунок 4 – Архитектура пользовательского интерфейса

Схема чётко показывает, какие элементы и действия активируются на уровне интерфейса, позволяя визуальнo понять структуру пользовательского опыта.

Так как визуальные элементы системы обладают простой и логичной структурой, освоить работу с приложением пользователю легко и удобно. Как видно из скриншотов, на каждой странице размещены формы и кнопки, соответствующие функциям, а данные представлены в виде таблиц.

Фронтенд, основанный на шаблонной системе Django, является эффективным, простым в структуре и надёжным техническим решением для организации взаимодействия между пользователем и приложением.

Бэкенд (серверная логика)

Серверная часть приложения — это основной логический центр, отвечающий за обработку действий пользователя и возврат соответствующих ответов. Этот компонент реализован на основе веб-фреймворка Django, что позволяет эффективно и просто организовать серверные процессы с использованием языка программирования Python.

Основная задача бэкенда — принимать запросы, поступающие с пользовательского интерфейса, обрабатывать их и возвращать соответствующие данные через базу данных. Например, когда пользователь добавляет новую финансовую транзакцию, запрос отправляется на сервер. Сервер проверяет данные, сохраняет их и возвращает на страницу обновлённую информацию.

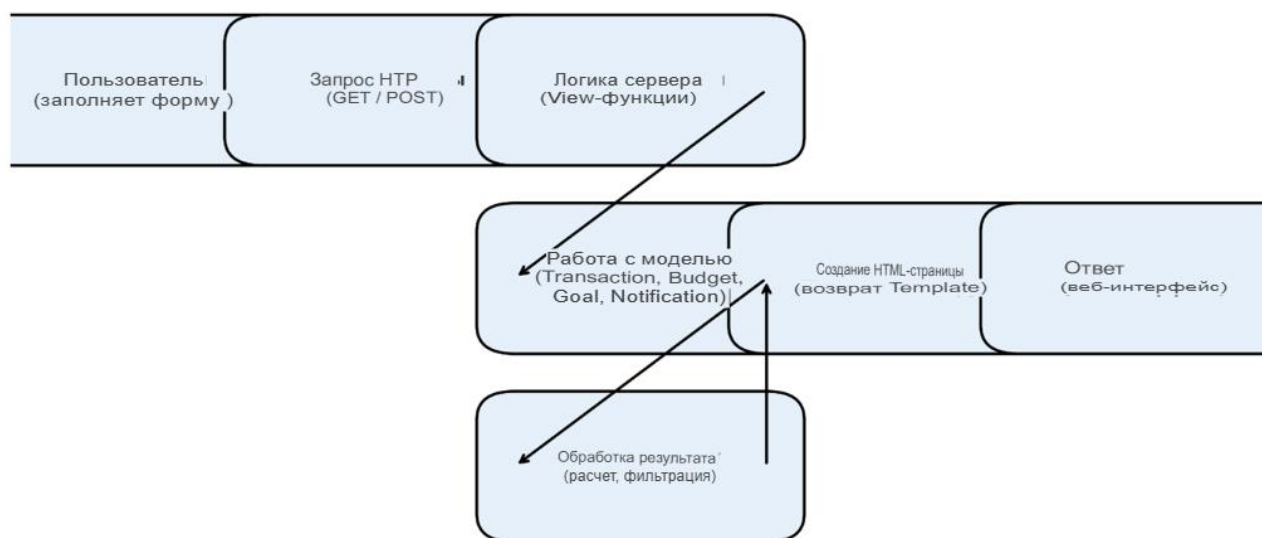


Рисунок 5 – Схема работы бэкенд-логики

Данная схема описывает основные этапы работы бэкенд (серверной) логики приложения для управления личными финансами. HTTP-запрос, поступающий от пользователя (например, отправка формы), сначала направляется в view-функции сервера. Здесь запрос обрабатывается и взаимодействует с соответствующими моделями (Transaction, Budget, Goal и др.). После обработки данных сервер формирует HTML-страницу на основе обработанной информации и возвращает её пользователю. Этот процесс реализован с использованием архитектуры MTV (Model–Template–View), схожей с MVC, во фреймворке Django.

В структуре проекта основная серверная логика выполняет следующие задачи:

- Регистрация и аутентификация пользователей (register_view, login_view, logout_view);
- Работа с финансовыми операциями: добавление, удаление транзакций, классификация по категориям (add_transaction, transaction_list, category_list);
- Установка и контроль бюджета (budget_view);
- Управление финансовыми целями (goal_create, goal_update, goal_delete);
- Вычисление аналитических данных (analytics_view, monthly_comparison_view);
- Отображение уведомлений и напоминаний (notification_list, модель Reminder).

Вся серверная логика ориентирована на аутентифицированного пользователя: каждая функция защищена декоратором `@login_required`, что обеспечивает безопасность данных.

Django использует архитектуру MTV (Model-Template-View), близкую к MVC (Model-View-Controller), где:

- **View** — функции, принимающие запросы пользователей;
- **Model** — таблицы, определяющие структуру данных;
- **Template** — HTML-страницы, возвращаемые в ответ.

Сервер получает информацию из базы данных, обрабатывает её и возвращает пользователю через HTML-страницы. Поскольку бизнес-логика сосредоточена на сервере, обеспечивается высокий уровень безопасности, надёжности и масштабируемости системы.

База данных

Во всех данных приложения для управления личными финансами используется реляционная база данных. Эта система предназначена для работы со структурированной информацией, такой как пользователи, транзакции, категории, цели и бюджеты, и управляется через ORM (Object-Relational Mapping) фреймворка Django. ORM автоматически преобразует объекты Python в таблицы базы данных.

Для работы с данными в приложении используются модели, определённые в файле `models.py`. Эти модели соответствуют конкретным таблицам и позволяют добавлять, обновлять, удалять или получать данные с помощью синтаксиса Python без написания SQL-запросов.

Основные модели, используемые в системе:

- **CustomUser** — расширенная модель пользователя (с электронной почтой, телефоном);
- **Transaction** — описывает каждую финансовую операцию (доход/расход, категория, сумма, время);
- **Category** — категории для транзакций;
- **Budget** — лимит расходов пользователя;
- **Goal** — финансовые цели (планируемая и накопленная сумма);
- **Notification** и **Reminder** — уведомления и напоминания.

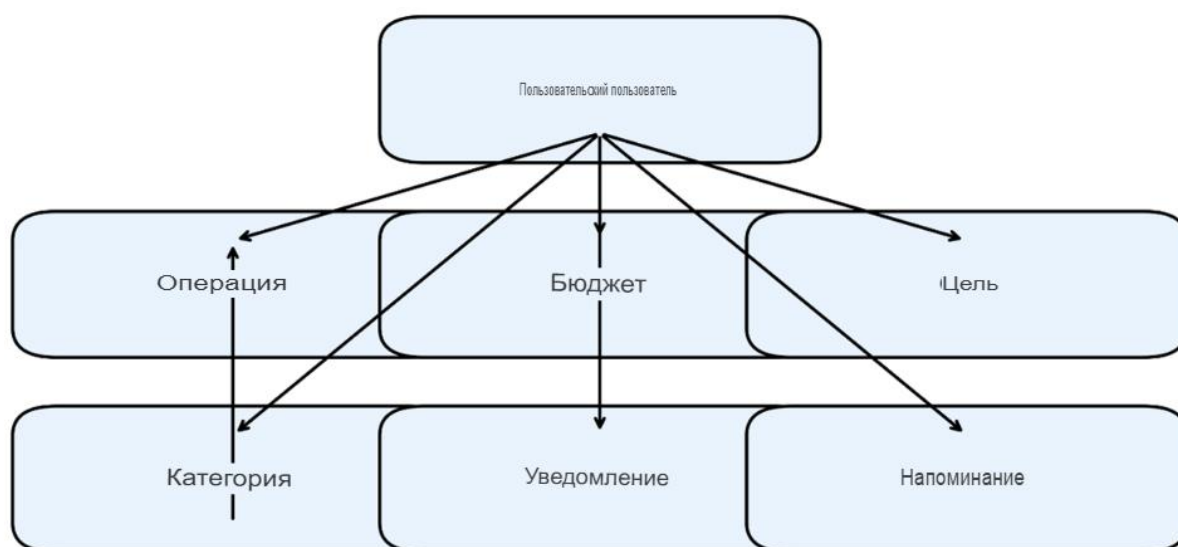


Рисунок 6 – Таблицы базы данных приложения и их взаимосвязь

Данная схема отражает структуру базы данных приложения и логические связи между таблицами. Таблица **CustomUser** играет центральную роль и связана со всеми основными сущностями: транзакциями, категориями, бюджетом, целями, уведомлениями и напоминаниями. Каждое поле типа **ForeignKey** указывает на зависимость между таблицами.

Таблица **Transaction** зависит как от пользователя, так и от категории. Такая архитектура обеспечивает масштабируемость системы, структурированное хранение данных и ведение индивидуальной информации для каждого пользователя.

Все модели наследуются от класса `models.Model` и автоматически регистрируются в базе данных как таблицы. Используемая в Django база данных SQLite удобно подходит для хранения данной структуры, однако при необходимости систему легко адаптировать под более мощные СУБД, такие как PostgreSQL или MySQL.

Ключевые преимущества:

- Безопасное хранение персональных данных пользователей;
- Привязка транзакций к конкретному пользователю (через `ForeignKey`);
- Поддержка целостности данных (через параметры `on_delete`);
- Автоматические временные метки (через `auto_now_add`, `timezone.now`);
- Поддержка агрегатных функций (`Sum`, `Count`).

Таким образом, база данных приложения централизованно хранит всю информацию, связанную с финансами и пользователями, обеспечивая надёжную и эффективную работу с данными.

Безопасность

Обеспечение информационной безопасности приложения — один из важнейших требований при его проектировании и реализации. Поскольку

приложение работает с личными финансовыми данными пользователей, необходимо защищать информацию на всех уровнях системы. В данном проекте реализованы встроенные механизмы аутентификации и авторизации Django, а также дополнительные меры и практики безопасности.

Аутентификация и авторизация

В системе создаются личные аккаунты пользователей, и доступ к основным функциям приложения (ввод транзакций, постановка целей, установка бюджета и т.д.) предоставляется только после аутентификации. Это реализуется с помощью декоратора `@login_required`, который проверяет, вошёл ли пользователь в систему, и при отсутствии авторизации блокирует доступ.

Хранение паролей

Пароли пользователей не сохраняются в открытом виде. Django автоматически шифрует их с помощью односторонней хеш-функции (PBKDF2, Argon2 или bcrypt). Это гарантирует, что даже при взломе базы данных восстановить настоящие пароли невозможно.

Сессии и cookies

При входе пользователя в систему запускается специальная сессия. Django хранит сессионную информацию на стороне сервера, а в браузер отправляется только cookie с идентификатором сессии. Таким образом, конфиденциальные данные не сохраняются в браузере.

Защита от CSRF

В приложении применяется встроенная защита от атак типа CSRF (межсайтовая подделка запроса). В HTML-формах автоматически добавляется специальное поле `csrf_token`, которое предотвращает обработку поддельных внешних запросов.

Целостность данных

Все основные модели проекта связаны через `ForeignKey` и ограничивают доступ к пользовательским данным, исключая возможность просмотра информации других пользователей. Каждый запрос ограничен в контексте конкретного пользователя через `request.user`.

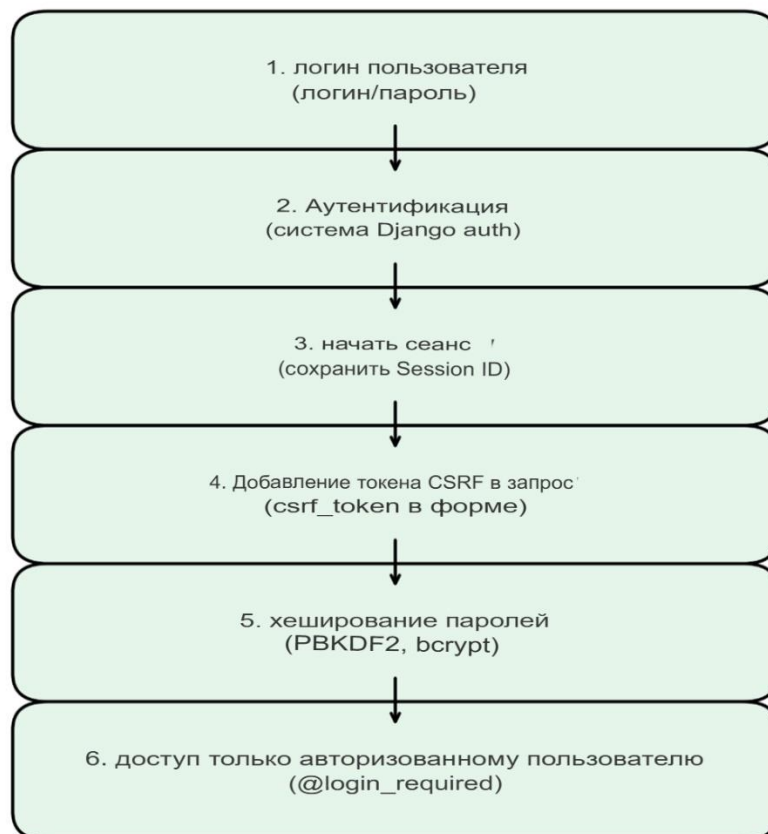


Рисунок 7 – Цепочка защиты данных в приложении

Схема поэтапно демонстрирует, как осуществляется защита пользовательских данных в системе. После того как пользователь входит в систему с помощью логина и пароля, система аутентификации Django проводит проверку. При успешном входе создаётся сессия, и на стороне сервера сохраняется специальный Session ID.

В каждой HTML-форме используется CSRF-токен, который защищает от поддельных внешних запросов. Пароли никогда не хранятся в открытом виде — они хэшируются с помощью безопасных алгоритмов. Кроме того, только аутентифицированным (вошедшим в систему) пользователям предоставляется доступ к работе с данными, что обеспечивается механизмом `@login_required`.

Все эти этапы усиливают безопасность системы и обеспечивают целостность и конфиденциальность пользовательских данных. Используемые в приложении механизмы безопасности — аутентификация, шифрование паролей, управление сессиями, защита от CSRF и ограничение доступа к пользовательским данным — делают систему надёжной и безопасной.

В целом, архитектура приложения спроектирована в соответствии с современными требованиями, а все её основные компоненты — фронтенд, бэкенд, база данных и безопасность — тесно взаимосвязаны. Используемая клиент-серверная модель, фреймворк Django и работа с реляционными базами данных обеспечивают надёжность и масштабируемость приложения.

Фронтенд предоставляет интерфейс для прямого взаимодействия с пользователем, в то время как бэкенд-логика выполняет основную функциональность системы и взаимодействует с базой данных. Безопасность данных реализована на всех уровнях архитектуры с помощью современных методов, направленных на защиту личной информации пользователя.

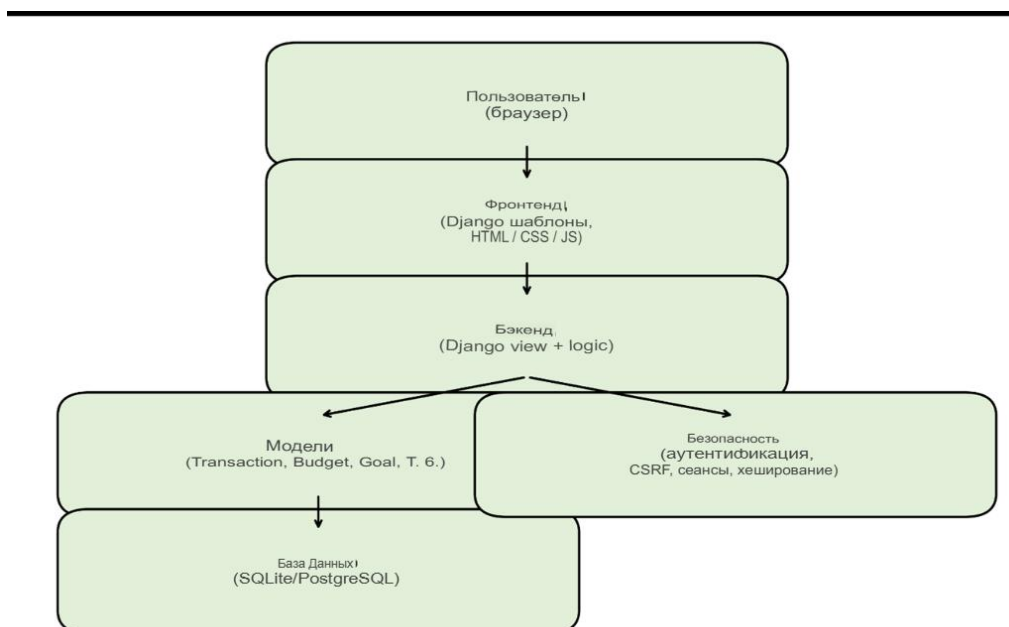


Рисунок 8 – Общая архитектурная структура приложения

Данная схема отображает основные компоненты системы — от пользователя до уровней фронтенда, бэкенда, моделей, базы данных и безопасности, а также их взаимосвязь. Такие комплексные архитектурные решения обеспечивают высокую функциональность приложения, удобство для пользователя и информационную безопасность, превращая систему в готовый продукт, пригодный для использования в реальной прикладной среде. Эта архитектура также служит прочной основой для последующего обновления, усовершенствования и масштабирования системы.

2.3 Пользовательский интерфейс и UX-дизайн

Эффективность финансового приложения определяется не только его функциональными возможностями, но и простотой и доступностью пользовательского интерфейса. Интерфейс, созданный в рамках этого проекта, основан на веб-версии и реализован с использованием шаблонной системы Django, при этом пользовательский опыт (UX) организован просто и логично.

Для того чтобы цифровой продукт был удобен пользователю, особое значение имеют элементы интерфейса и UX-дизайна. Пользовательский интерфейс (User Interface – UI) и пользовательский опыт (User Experience – UX) — это ключевые показатели доступности и понятности приложения. Веб-версия приложения для управления личными финансами реализована на шаблонах Django и создана с опорой на принципы простоты, наглядности и структурности.

Интерфейс основан на трёх основных принципах:

- **Простота** — на каждой странице отображаются только основные элементы и функции. Пользователь не отвлекается на лишнюю информацию и может сосредоточиться на управлении своими финансами.

- **Понятность** — названия форм и кнопок даны на казахском языке, чёткие и соответствуют своему назначению. Например, действия «Қіріс қосу» (Добавить доход), «Шығындарды қарау» (Просмотр расходов), «Мақсат құру» (Создать цель) расположены в логичном порядке.

- **Последовательность** — все страницы оформлены в едином стиле, что облегчает навигацию по системе.

Интерфейс обеспечивает непрерывное взаимодействие между пользователем и приложением. Его структура разработана на понятном языке, элементы управления и необходимая информация логично организованы. Каждая страница направлена на выполнение конкретной задачи: добавление транзакции, создание цели, управление бюджетом. Такая структура позволяет быстро освоить систему и эффективно с ней работать.

UX-дизайн приложения делает акцент на удобстве пользователя. Все действия пользователя реализованы просто, последовательно и логично. Формы, списки и кнопки на интерфейсе визуально выделены. Например, кнопки «Добавить», «Редактировать», «Удалить» окрашены в определённые цвета, соответствующие их назначению, что снижает вероятность ошибок.

В UX-дизайне учтены следующие аспекты:

- **Быстрый доступ** — нужная информация представляется кратчайшим путём, без лишних переходов (например, статистика по доходам и расходам отображается на главной странице).

- **Интерактивность** — введённые данные сразу обрабатываются и отображаются пользователю с обратной связью.

- **Наглядность кнопок и форм** — каждая кнопка отличается цветом и размером, что облегчает управление с мобильных устройств.

- **Адаптивность** — структура интерфейса подстраивается под различные размеры экранов.

Вводимые данные обрабатываются моментально, а результат отображается в наглядной форме. Скорость обратной связи подтверждает высокое качество UX-дизайна. Также предусмотрены аналитические страницы для пользователей. На этих страницах визуализированы данные о финансовых транзакциях в виде диаграмм и таблиц, позволяя пользователю контролировать свой бюджет и расходы.

Структура интерфейса адаптирована под различные устройства. Несмотря на то, что приложение построено на веб-платформе, его структура обеспечивает удобный внешний вид и на мобильных устройствах. При использовании адаптивных фреймворков, таких как HTML/CSS и Bootstrap, система автоматически подстраивается под размер экрана, предлагая пользователю комфортный интерфейс.

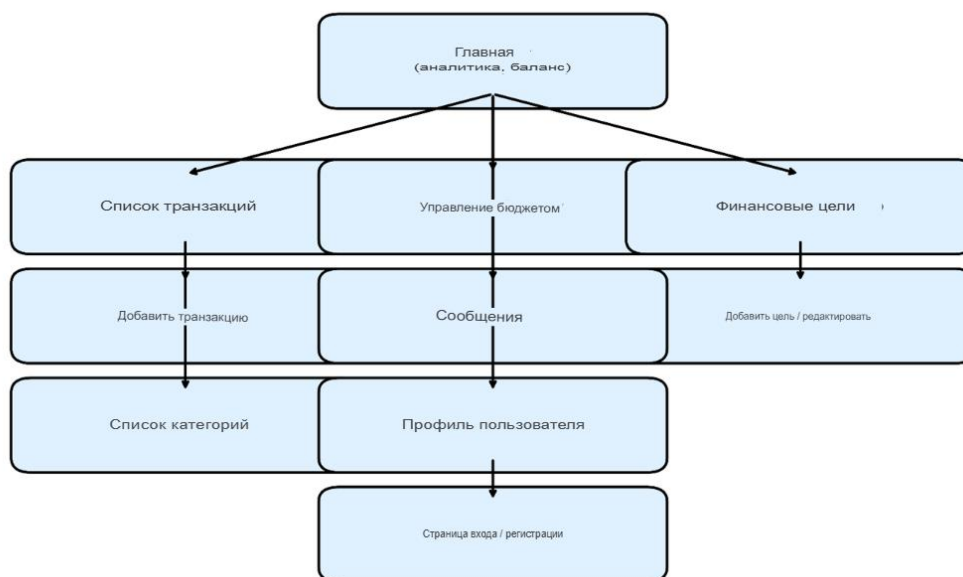


Рисунок 9 – Структура интерфейса финансового приложения

Схема отображает элементы интерфейса мобильного финансового приложения и их взаимосвязь. Главная страница выступает в качестве центрального узла системы, откуда осуществляется переход к основным разделам — транзакции, бюджет, цели, уведомления и профиль.

Каждая функция включает в себя дополнительные подстраницы: например, через страницу транзакций можно добавить операцию или управлять категориями, а в разделе целей предусмотрено редактирование целей. Также через профиль пользователя осуществляется вход в систему или регистрация.

Интерфейс организован логично и последовательно, что позволяет пользователю легко и быстро взаимодействовать с приложением.

Хотя в текущей версии ещё не реализованы светлая и тёмная темы оформления, в будущем добавление данной функции повысит адаптивность приложения и удовлетворит предпочтения различных пользователей. Это рассматривается как одно из возможных направлений улучшения UX-дизайна.

Таким образом, интерфейс и UX-дизайн этого приложения направлены на то, чтобы обеспечить понятную, логичную, эффективную и удобную работу пользователя. Структура системы и элементы управления организованы по принципам простоты и интуитивной понятности. В результате пользователь может управлять своими финансами без технических затруднений, быстро и легко получая доступ к нужной информации.

2.4 Выбор инструментов и технологий разработки

При разработке данной мобильного-приложения для управления личными финансами были использованы современные, надёжные и гибкие технологии. Выбранные инструменты обеспечивают масштабируемость системы, безопасность данных и эффективность процесса разработки.

В процессе разработки структура проекта была организована по модульному принципу. Это позволило каждому компоненту (пользовательский интерфейс, бизнес-логика, работа с базой данных, безопасность) функционировать независимо, а при необходимости — обновляться и расширяться отдельно.

Серверная часть приложения написана с использованием языка Python и фреймворка Django, где реализована вся основная логика. Этот фреймворк включает в себя развитую ORM-систему, встроенные механизмы аутентификации и внутренние средства обеспечения безопасности.

Фронтенд реализован с помощью шаблонной системы Django, а визуальный интерфейс — с использованием HTML и CSS. Простой и интуитивно понятный интерфейс создаёт удобные условия для взаимодействия пользователя с системой.

В качестве базы данных в проекте использовалась SQLite — эффективная и неприхотливая реляционная СУБД для малых и средних систем. При этом структура приложения легко адаптируется под промышленные СУБД, такие как PostgreSQL.

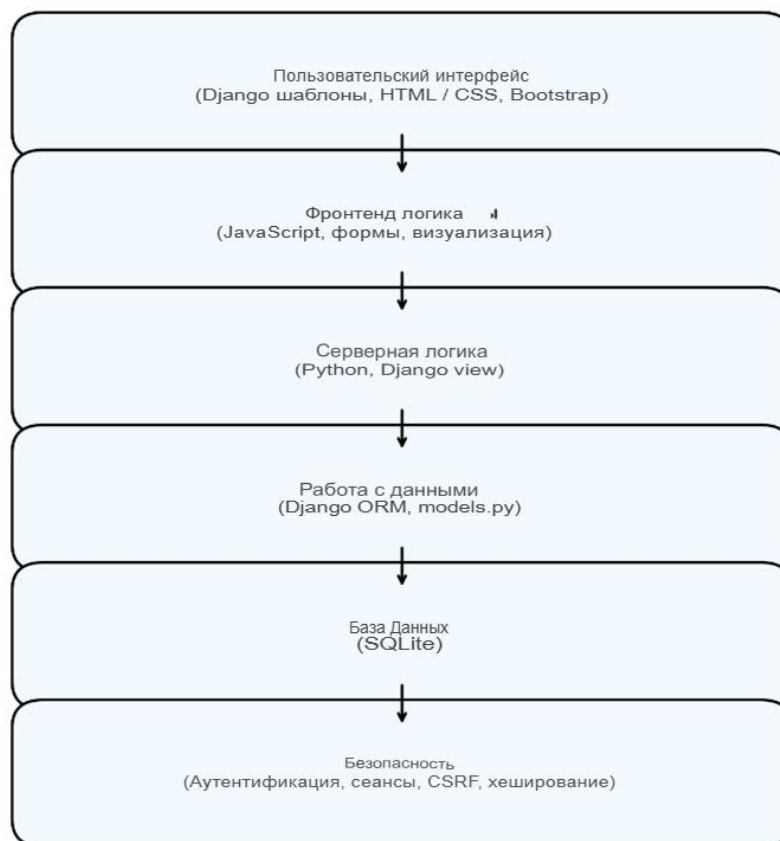


Рисунок 10 – Структура технологического стека приложения

Данная диаграмма пошагово отображает все основные технологические уровни (стеки) веб-приложения. Каждый слой выполняет определённую функцию и служит основой стабильной и надёжной работы всей системы.

В качестве основного языка программирования в проекте был выбран Python. Python — один из самых популярных, многоцелевых и открытых языков программирования на сегодняшний день. Его простой синтаксис, высокая читаемость и большое сообщество делают его идеальным выбором для разработки веб-приложений.

Вместе с фреймворком Django язык Python позволил реализовать всю серверную логику приложения, механизмы работы с пользователями, обработку данных и меры безопасности. С помощью Django ORM взаимодействие с базой данных было выполнено просто и надёжно.

Преимущества языка Python:

- Простой и легко читаемый синтаксис;
- Множество встроенных библиотек и модулей;
- Поддержка мощных веб-фреймворков, таких как Django и Flask;
- Широкое применение в безопасности, тестировании, визуализации и науке о данных.

Выбор Python полностью соответствовал требованиям проекта и позволил создать надёжную, устойчивую и масштабируемую архитектуру приложения.

Django как основной фреймворк

Для разработки проекта был выбран веб-фреймворк Django. Это полнофункциональный, высокоуровневый фреймворк на языке Python, ориентированный на быстрое и безопасное создание веб-приложений. Django использует архитектурную модель MTV (Model–Template–View), которая близка к классической MVC.

Возможности Django:

- Встроенная система аутентификации — регистрация пользователей, вход, управление правами доступа;
- ORM (Object-Relational Mapping) — работа с базой данных без необходимости писать SQL-запросы;
- Шаблонная система — динамическое формирование HTML-страниц на сервере;
- Административная панель — автогенерируемый интерфейс для работы с моделями;
- Безопасность — встроенная защита от CSRF, XSS и SQL-инъекций;
- Модульность — удобная организация проекта по отдельным приложениям (apps).

Благодаря Django были достигнуты преимущества в виде сокращения времени разработки, структурированной архитектуры и расширяемости. Обширное сообщество и качественная документация упростили поиск решений и технической поддержки.

Работа с базой данных: SQLite

Для проекта была выбрана система управления базами данных **SQLite**. Это лёгкая, серверонезависимая реляционная СУБД, все данные в которой хранятся в одном .db файле. Django по умолчанию поддерживает SQLite, и она идеально подходит для небольших и средних проектов.

Преимущества SQLite:

- Не требует дополнительной настройки — работает по принципу «установил и используй»;
- Быстрая и простая в использовании;
- Отлично подходит для тестирования и ранних стадий разработки;
- Полностью интегрирована в среду Python/Django;
- Малый размер файлов и кроссплатформенность.

Django ORM позволяет описывать модели в виде Python-классов, автоматически превращая их в таблицы базы данных. Такой подход упрощает работу с данными и обеспечивает структурную целостность проекта.

При необходимости SQLite легко заменяется на промышленную СУБД, такую как PostgreSQL или MySQL, что позволяет сохранить потенциал масштабирования.

Среда разработки

Для основного процесса разработки был использован **Visual Studio Code (VS Code)** — многофункциональная и расширяемая IDE, прекрасно подходящая

для работы с Python и Django. Простота интерфейса, поддержка плагинов и встроенный терминал сделали процесс разработки удобным и эффективным.

С помощью расширений в VS Code обеспечивалась автоматическая проверка синтаксиса Python, ускоренное написание кода через IntelliSense, а встроенная поддержка Git позволяла управлять версией проекта.

В некоторых задачах, особенно при работе с моделями и базой данных, использовалась также **PyCharm Community Edition** — мощная IDE для профессиональной разработки на Python.

Интерфейс приложения

Пользовательский интерфейс (UI) приложения был реализован с помощью **шаблонной системы Django (Django Template Language)**, которая позволяет динамически формировать HTML-страницы на сервере, объединяя Python-код и HTML в единый шаблон.

Для оформления интерфейса использовались **HTML, CSS** и фреймворк **Bootstrap** — одна из самых популярных CSS-библиотек, позволяющая создавать адаптивный и визуально согласованный интерфейс. Это обеспечило хорошую поддержку различных экранов и мобильных устройств.

Простые элементы **JavaScript** использовались для проверки форм, обработки действий кнопок и отображения динамических компонентов, что улучшило отзывчивость интерфейса.

Запуск и размещение приложения

На этапе разработки приложение запускалось на **локальном сервере (localhost)**, используя встроенный сервер Django, что позволяло быстро тестировать и отлаживать систему.

Для размещения в производственной среде рассматривались следующие платформы:

- **Heroku** — удобная облачная платформа для развёртывания Django-приложений с поддержкой PostgreSQL;
- **PythonAnywhere** — подходящая для небольших проектов платформа с веб-интерфейсом управления;
- **VPS или Linux-серверы** — для крупных систем с возможностью гибкой настройки и полного контроля.

Дополнительные технологии для деплоя:

- **Gunicorn или uWSGI** — для запуска Django в боевой среде;
- **Nginx или Apache** — в качестве веб-серверов;
- **GitHub Actions или другие CI/CD-инструменты** — для автоматизации процесса деплоя.

Использованные технологии — язык Python, фреймворк Django, база данных SQLite, шаблонная система Django, Visual Studio Code — обеспечили быструю, надёжную и структурированную разработку приложения. Каждый компонент был выбран с учётом требований проекта и позволил обеспечить безопасность, расширяемость и техническую готовность к дальнейшему размещению в рабочей среде.

3 Реализация и тестирование мобильного приложения

3.1 Этапы разработки

Разработка мобильного приложения — это не просто процесс написания кода, а комплексная работа, состоящая из нескольких структурных и взаимосвязанных этапов. Каждый этап формируется в соответствии с целями проекта и направлен на обеспечение функциональной полноты, надёжности и удобства системы для пользователя. Все стадии — от проектирования до тестирования и релиза — были тщательно спланированы и реализованы последовательно, шаг за шагом. В данном разделе описаны основные этапы процесса разработки приложения: от выявления и анализа требований до проектирования интерфейса, построения базы данных, реализации ключевых функций, тестирования и подготовки системы к выпуску. Каждый этап был ориентирован на конкретные цели и результаты, что способствовало созданию качественной системы.

Первый и один из самых важных этапов разработки приложения — всестороннее изучение и анализ требований к системе. На этом этапе были определены цели проекта, целевая аудитория, ключевые функциональные возможности и технические ограничения.

Вначале изучались основные потребности пользователей и их повседневное финансовое поведение. Были проанализированы современные мобильные и мобильного-приложения для ведения личного бюджета, рассмотрены их функциональные возможности, удобство для пользователей, структура интерфейса и эффективность с точки зрения пользовательского опыта (UX). На основе этого анализа был определён перечень функций, которые необходимо включить в систему.

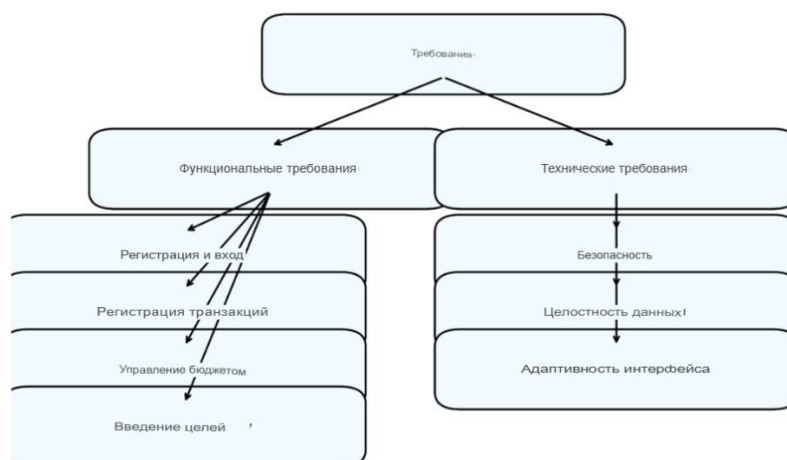


Рисунок 11 – Диаграмма классификации требований к приложению

Основное внимание было уделено следующим функциональным возможностям:

- Регистрация пользователя и безопасный вход в систему;
- Регистрация финансовых операций (доходы/расходы);
- Группировка операций по категориям;
- Установка и контроль бюджетных ограничений;
- Постановка финансовых целей и отслеживание прогресса накоплений;
- Система уведомлений и напоминаний;
- Просмотр аналитических диаграмм и отчетов.

Также были определены технические требования системы:

- Безопасность (защита паролей, авторизация);
- Целостность пользовательских данных;
- Модульная структура и возможность расширения;
- Интерфейс, адаптированный под веб-версию, и простая навигация.

В качестве результата данного этапа было подготовлено исходное техническое задание (ТЗ) проекта. В нём были описаны функциональные блоки, их взаимосвязь и последовательность разработки. Тщательный анализ требований заложил основу проекта и определил направление всех последующих этапов.

Создание UI/UX прототипов

В процессе разработки мобильного приложения важную роль сыграл этап проектирования пользовательского интерфейса (UI) и пользовательского опыта (UX). Этот этап определяет структуру системы, взаимодействие пользователя с системой и логическое представление интерфейса.

При создании UI/UX-прототипов особое внимание уделялось удобству использования приложения, простоте и понятности интерфейса.

Работа над прототипами проводилась в следующей последовательности: на первом этапе определялись основные страницы системы и их взаимосвязь. Планировалось, на каких страницах будут выполняться действия пользователя.

Основные страницы:

- Главная страница (аналитика, баланс, общая статистика);
- Страница транзакций (список доходов и расходов);
- Страница добавления транзакции (форма заполнения);
- Страница управления бюджетом;
- Страница финансовых целей;
- Профиль пользователя;
- Страницы входа и регистрации.

При создании дизайна логично размещались визуальные элементы:

- Кнопки (Button)
- Таблицы (Table)
- Меню (Menu)
- Формы (Form)
- Диаграммы (Chart)

Каждая кнопка в интерфейсе выделялась цветом и размером. Поля ввода

информации располагались просто и понятно для пользователя.

На первом этапе прототипирование проводилось в виде простых схем (Wireframe) на бумаге или в графическом виде. Позже структура была реализована с использованием **шаблонной системы Django** в виде реальных HTML-страниц. Адаптивный дизайн создавался с помощью **CSS** и **Bootstrap**.

При разработке UI/UX-прототипа логическая структура интерфейса была построена так, чтобы:

- пользователь мог быстро находить основную информацию;
- свободно перемещаться по системе и легко выполнять нужные действия;
- навигация была простой и последовательной.

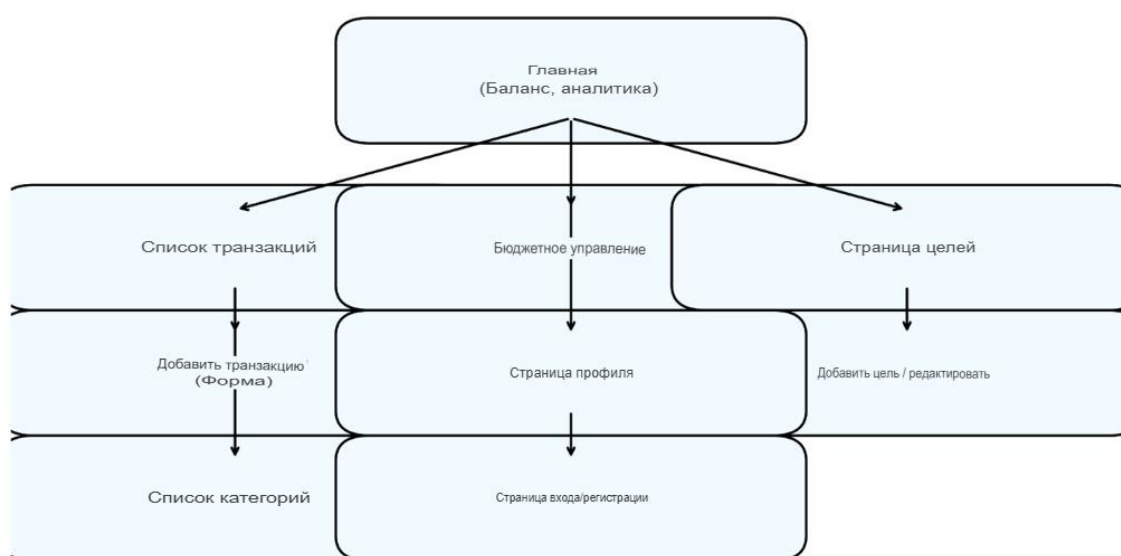


Рисунок 12 – Структурная схема интерфейса приложения

Схема отображает основные страницы приложения и их взаимосвязь. Показано, через какие страницы реализуются те или иные функциональные модули и какова их логическая последовательность.

Этап создания UI/UX-прототипов позволил организовать интерфейс системы так, чтобы он был максимально удобным и интуитивно понятным для пользователя. Структурные схемы и взаимосвязь между страницами, разработанные на этом этапе, способствовали эффективной работе системы и улучшению пользовательского опыта.

Настройка базы данных

Настройка базы данных в процессе разработки мобильного приложения — один из важнейших технических этапов проекта. На данном этапе была определена структура всех данных, хранящихся в системе, спроектированы их связи и реализованы средствами фреймворка Django.

Работы по настройке базы данных проводились в следующем порядке:

В основном конфигурационном файле проекта **settings.py** были заданы параметры базы данных. Для упрощения разработки была выбрана СУБД **SQLite**.

```
DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.sqlite3',
        'NAME': BASE_DIR / 'db.sqlite3',
    }
}
```

Все таблицы базы данных описывались в виде Python-классов с использованием **ORM Django**.

Основные модели проекта:

CustomUser — для хранения информации о пользователе:

```
class CustomUser(AbstractUser):
    phone_number = models.CharField(max_length=15, null=True, blank=True)
```

Transaction — для хранения транзакций:

```
class Transaction(models.Model):
    user = models.ForeignKey(CustomUser, on_delete=models.CASCADE)
    amount = models.DecimalField(max_digits=10, decimal_places=2)
    category = models.ForeignKey(Category, on_delete=models.SET_NULL,
null=True)
    type = models.CharField(max_length=10, choices=[('income', 'Доход'),
('expense', 'Расход')])
    created_at = models.DateTimeField(auto_now_add=True)
```

Прочие модели:

- **Category** — категории транзакций
- **Goal** — финансовые цели
- **Budget** — бюджет
- **Notification** — уведомления
- **Reminder** — напоминания

Между всеми моделями были установлены связи через **ForeignKey**.

После создания моделей структура данных была сохранена в БД с помощью миграций:

```
python manage.py makemigrations
python manage.py migrate
```

Данные привязывались к конкретному пользователю:

```
transactions = Transaction.objects.filter(user=request.user)
```

Настройка базы данных обеспечила структуру для всех модулей системы. С помощью ORM Django процесс работы с таблицами и данными стал значительно проще, а база данных получила гибкую и расширяемую архитектуру.

Реализация функциональности

Веб-приложение для управления личными финансами было реализовано на языке **Python** с использованием фреймворка **Django**, на основе архитектурной модели MTV (Model - Template - View), аналогичной MVC.

Регистрация пользователей и вход в систему

Модель:

```
class CustomUser(AbstractUser):
```

```
    phone_number = models.CharField(max_length=15, null=True, blank=True)
```

View:

```
def register_view(request):
```

```
    form = UserCreationForm()
```

```
    if request.method == 'POST':
```

```
        form = UserCreationForm(request.POST)
```

```
        if form.is_valid():
```

```
            form.save()
```

```
            return redirect('login')
```

```
    return render(request, 'users/register.html', {'form': form})
```

URL:

```
urls.py: path('register/', register_view, name='register')
```

Шаблон: register.html

Работа с транзакциями (доходы/расходы)

Модель:

```
class Transaction(models.Model):
```

```
    user = models.ForeignKey(CustomUser, on_delete=models.CASCADE)
```

```
    amount = models.DecimalField(max_digits=10, decimal_places=2)
```

```
    category = models.ForeignKey(Category, on_delete=models.SET_NULL,
null=True)
```

```
    type = models.CharField(max_length=10, choices=[('income', 'Доход'),
('expense', 'Расход')])
```

```
    created_at = models.DateTimeField(auto_now_add=True)
```

View:

```
def add_transaction(request):
```

```
    form = TransactionForm()
```

```
    if request.method == 'POST':
```

```

form = TransactionForm(request.POST)
if form.is_valid():
    transaction = form.save(commit=False)
    transaction.user = request.user
    transaction.save()
    return redirect('transaction_list')
return render(request, 'transactions/add_transaction.html', {'form': form})

```

Работа с финансовыми целями

Модель:

```

class Goal(models.Model):
    user = models.ForeignKey(CustomUser, on_delete=models.CASCADE)
    title = models.CharField(max_length=100)
    target_amount = models.DecimalField(max_digits=10, decimal_places=2)
    saved_amount = models.DecimalField(max_digits=10, decimal_places=2,
default=0)

```

View:

```

def goal_create(request):
    form = GoalForm()
    if request.method == 'POST':
        form = GoalForm(request.POST)
        if form.is_valid():
            goal = form.save(commit=False)
            goal.user = request.user
            goal.save()
            return redirect('goal_list')
    return render(request, 'goals/goal_create.html', {'form': form})

```

Установка и контроль бюджета

Модель:

```

class Budget(models.Model):
    user = models.ForeignKey(CustomUser, on_delete=models.CASCADE)
    limit = models.DecimalField(max_digits=10, decimal_places=2)

```

View:

```

def budget_view(request):
    budget = Budget.objects.filter(user=request.user).first()
    return render(request, 'budget/budget_view.html', {'budget': budget})

```

Разработка мобильного приложения велась поэтапно. На каждом этапе уточнялась структура системы и реализовывались необходимые функции с акцентом на удобство пользователя. Были полностью реализованы настройка базы данных, проектирование интерфейса и программная реализация ключевых возможностей приложения.

3.2 Реализация основных функций

В этом разделе подробно описано, как были реализованы ключевые функции мобильного приложения, логика их работы и конечные результаты. В ходе разработки каждая функция создавалась с целью обеспечить пользователю удобное и эффективное управление повседневными финансовыми операциями.

Каждая функция была реализована в соответствии с поставленными требованиями, а основой разработки стали язык программирования **Python** и фреймворк **Django**. Благодаря этому удалось эффективно организовать серверную часть логики, обработку и хранение данных, а также отображение их в интерфейсе.

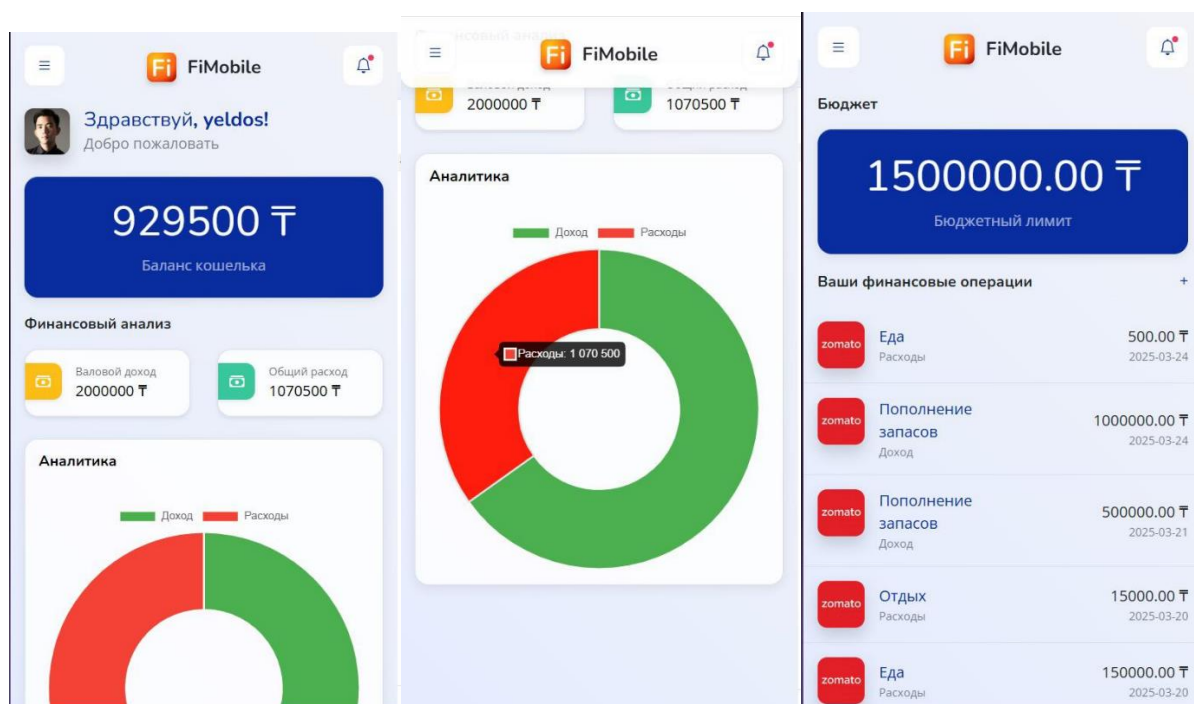


Рисунок 13 – Страница с возможностями приложения

Функциональные возможности мобильного приложения были разделены на несколько ключевых модулей:

- **Учёт доходов и расходов** — пользователь получил возможность вводить и отслеживать свои финансовые операции;
- **Финансовая аналитика** — визуальный обзор структуры расходов и доходов, оценка финансового состояния с помощью диаграмм и отчётов;
- **Напоминания** — уведомления пользователю о необходимости своевременной оплаты, с отображением внутри приложения или отправкой по email;

- **Финансовые цели и планы** — возможность для пользователя накопить средства за определённый период, создавать цели и отслеживать прогресс их достижения.

Каждая функция реализована на отдельной интерфейсной странице. На этих страницах предусмотрены формы для добавления операций, ввода данных, их обработки и отображения результатов. Все введённые данные сохраняются в базе данных и используются для построения аналитических отчётов.

Результаты работы функций отображаются на интерфейсных страницах приложения. Эффективность и результативность каждого модуля иллюстрировались скриншотами, наглядно демонстрирующими, как работает система. Такой подход позволил полноценно описать функциональные возможности разработанного приложения, логику его работы и уровень предоставляемых пользователю услуг, усилив практическую ценность проекта.

3.2.1 Учёт доходов и расходов

Одна из основных и наиболее часто используемых функций мобильного приложения для управления финансами — это учёт доходов и расходов. Данная функция позволяет пользователю регистрировать повседневные финансовые операции, распределять их по типу и категориям, а также автоматизировать процесс ведения личного бюджета.

Пользователь может ввести в систему все свои доходы и расходы через специальную форму. При вводе данных указываются следующие параметры: тип операции (доход или расход), категория, сумма и дата проведения. Введённые операции автоматически сохраняются в базе данных и впоследствии используются для общего финансового анализа.

В приложении предусмотрены два способа регистрации финансовых операций:

- **Первый** — ручной ввод данных. Пользователь самостоятельно заполняет форму, вводя конкретную доходную или расходную операцию. Форма простая и удобная для пользователя.
- **Второй** — добавление данных с использованием шаблонов. Особенно полезно для регулярных платежей или поступлений. Например, операции по зарплате, коммунальным платежам, подпискам можно добавить одним нажатием кнопки, используя заранее сохранённый шаблон.

Кроме того, реализована функция **поддержки повторяющихся платежей**. Пользователь может задать операцию с повторением (еженедельно, ежемесячно, ежегодно), что экономит время и способствует автоматизации финансовой дисциплины.

Все введённые операции отображаются в специальном **списке транзакций**, где указаны дата, сумма, категория и тип операции. При необходимости пользователь может отредактировать или удалить любую запись.

Функция учёта финансовых операций позволяет пользователю держать личные финансы под контролем и развивать финансовую грамотность. Категоризация данных и поддержка повторяющихся платежей значительно повышают функциональную эффективность системы.

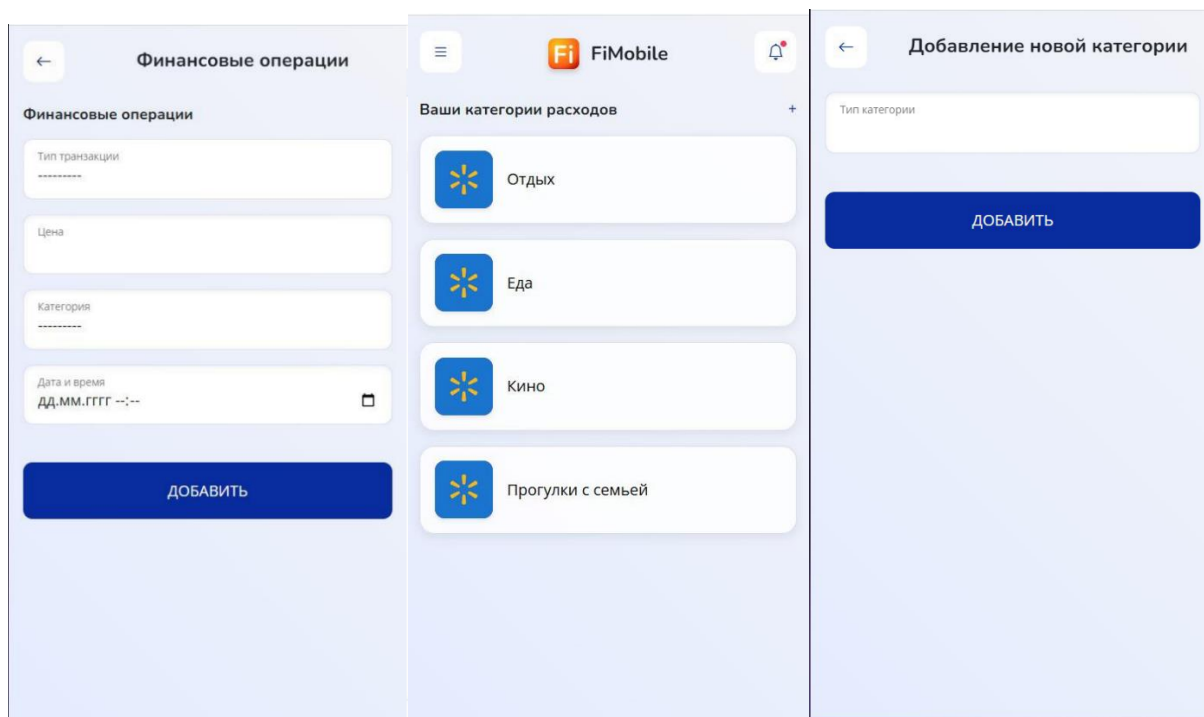


Рисунок 14 – Страница категории и операции доходов и расходов

На рисунке представлена специальная страница приложения для ввода доходов и расходов. Здесь пользователь может ввести полную информацию о своих финансовых операциях. Интерфейс организован просто и понятно, что позволяет легко заполнить форму без дополнительных пояснений.

В первую очередь в форме предлагается выбрать тип операции — **доход** или **расход**. Это помогает пользователю точно определить направление транзакции. Далее следует поле для ввода **суммы операции**. Сумма вводится в числовом формате, система автоматически обрабатывает значение и при необходимости производит его проверку.

Также пользователь может выбрать **категорию операции**. Категории предлагаются из заранее подготовленного списка, но также доступна возможность добавить новую категорию при необходимости. Это особенно удобно для структурирования расходов и доходов.

Есть возможность выбрать **дату операции** — пользователь может указать желаемую дату или сохранить операцию с текущей датой. Кроме того, если операция является **повторяющейся**, система автоматически будет вносить её в соответствии с заданным интервалом.

После заполнения формы пользователь может сохранить операцию в базе данных с помощью кнопки **«Сохранить»**. Все добавленные операции

отображаются в **списке транзакций**, где указываются дата, сумма, категория и тип операции. Таким образом, пользователь получает доступ к полной истории своих финансовых операций.

Представленный скриншот иллюстрирует одну из самых часто используемых и важных страниц системы. Пользователь может быстро и удобно вносить свои операции и эффективно управлять личными финансами.

3.2.2 Финансовая аналитика

Финансовая аналитика — одна из ключевых функций мобильного приложения. Этот раздел позволяет проводить полноценный анализ финансовых операций пользователя, отслеживать структуру доходов и расходов, а также оценивать эффективность ведения личного бюджета.

Аналитические отчёты и визуальные диаграммы помогают пользователю чётко понимать своё финансовое положение. В системе автоматически формируется статистика, которая представляется в наглядной форме с помощью графиков и диаграмм.

Функциональность финансовой аналитики включает в себя следующие возможности:

- Автоматическая генерация **ежемесячных и годовых отчётов**;
- Визуализация **распределения доходов и расходов по категориям** в виде диаграмм;
- **Мониторинг динамики общего баланса** за определённый период;
- Наглядное отображение **суммарных затрат по каждой категории**.

На аналитической странице пользователь может **отфильтровать данные** по нужным параметрам. Например, есть возможность просматривать расходы за определённый месяц или только операции по доходам.

Круговая диаграмма (Pie chart) визуализирует распределение доходов и расходов по категориям. Каждый цветной сектор представляет определённую категорию, что наглядно показывает, на какие области пользователь тратит больше всего средств.

График отражает динамику баланса пользователя за определённый период. Он показывает рост или снижение доходов и расходов, позволяя пользователю анализировать своё финансовое состояние по месяцам или дням.

Эта функция помогает пользователю отслеживать свою финансовую дисциплину, выявлять лишние траты и повышать эффективность личного бюджета. Благодаря простому и понятному интерфейсу, страница аналитики является основным инструментом для оценки финансового состояния.

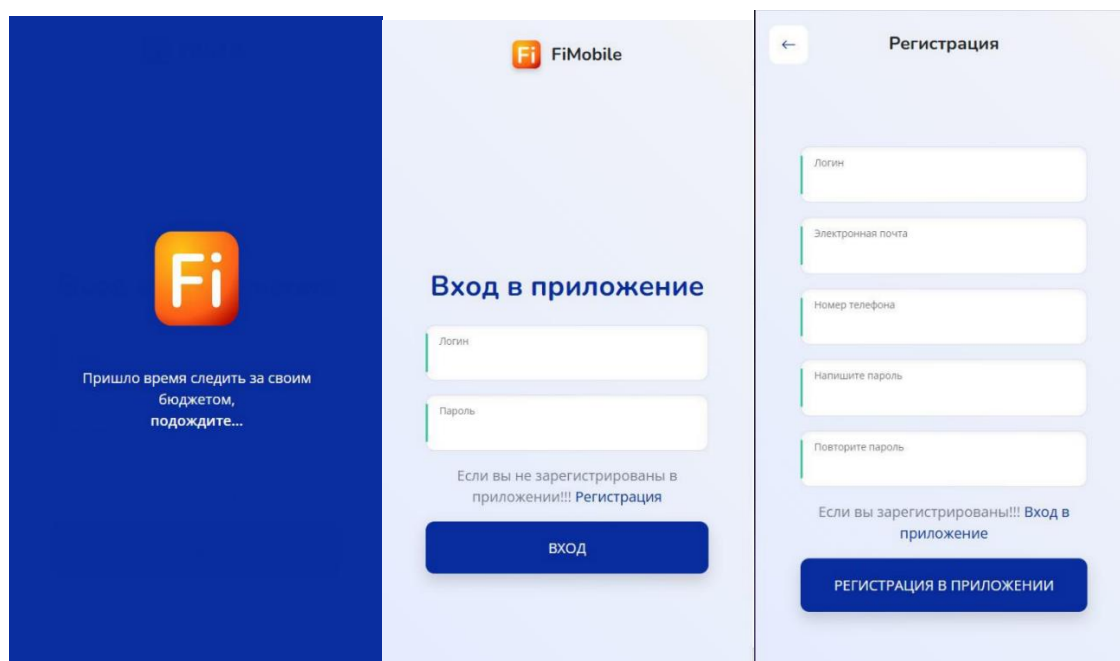


Рисунок 15 – Страница авторизации

На рисунке представлен интерфейс страницы авторизации. **Страница авторизации** — один из ключевых компонентов приложения, обеспечивающий **безопасный доступ** пользователя к его личным финансовым данным. Она предназначена для идентификации и аутентификации пользователя перед получением доступа к системе.

Безопасность

Все данные, передаваемые через форму, защищаются стандартными средствами Django — используется защита от CSRF-атак и шифрование паролей через хеш-функции (PBKDF2, Argon2 и др.). Авторизация возможна **только для зарегистрированных пользователей** — проверка осуществляется с использованием встроенной системы аутентификации Django.

Логика работы:

1. Пользователь вводит логин и пароль.
2. Система проверяет данные.
3. При успешной проверке создаётся **сессия**, и пользователь перенаправляется на **главную страницу** (например, аналитику или список транзакций).
4. В случае ошибки отображается сообщение: *"Неверное имя пользователя или пароль."*

Страница авторизации обеспечивает вход в систему и играет важную роль в защите персональных данных. Её простая структура и встроенные механизмы безопасности делают процесс авторизации надёжным и удобным для пользователей.

3.2.3 Напоминания

Функция напоминаний является одной из важных возможностей мобильного приложения. Она помогает пользователю поддерживать финансовую дисциплину, вовремя выполнять платежи и не забывать о своих финансовых планах.

В системе предусмотрена возможность настройки напоминаний о платежах. Пользователь может установить напоминание для определённой финансовой операции или обязательства. Напоминания настраиваются вручную — указываются дата и частота повторения (например, раз в неделю, ежемесячно и т. д.).

В приложении реализованы два способа уведомлений:

- Показ **внутри приложения** (Push-уведомления);
- Отправка **по электронной почте**.

Преимущество этого функционала в том, что пользователь может получить напоминание даже без входа в приложение — уведомление появится на экране или придёт на почту.

Все установленные напоминания хранятся на отдельной странице, где их можно изменить или удалить. Когда приближается срок оплаты или наступает дата следующего платежа, система автоматически отправляет соответствующее уведомление.

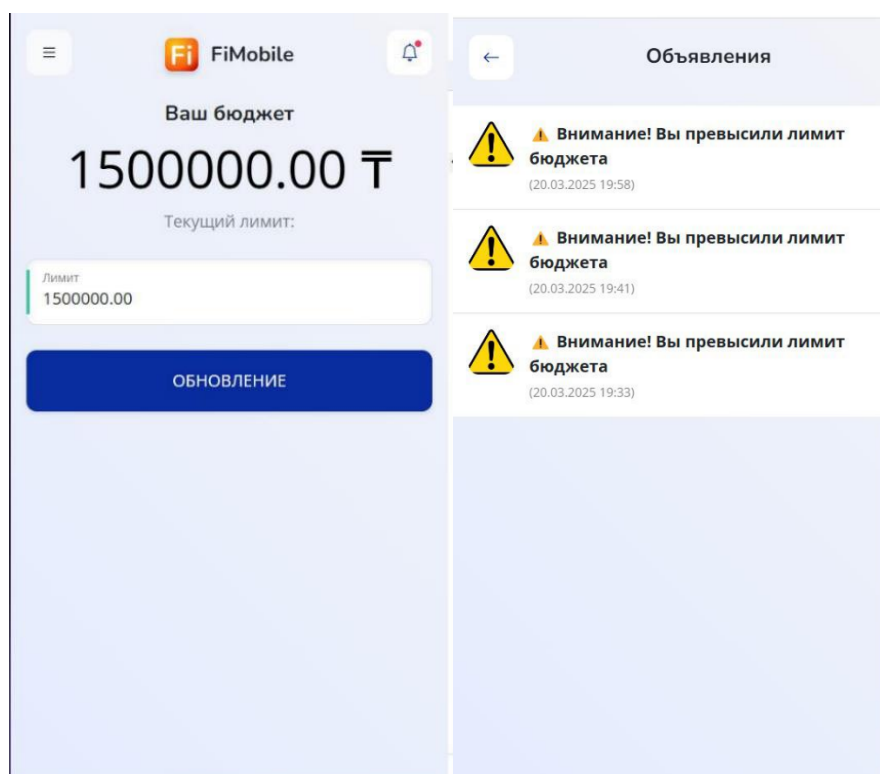


Рисунок 16 – Страница настройки напоминания

На рисунке показан процесс установки напоминания о платеже пользователем. Пользователь может легко задать название платежа, дату и частоту повторения, чтобы установить напоминание. Когда наступает установленное время, уведомление автоматически появляется на экране устройства или отправляется на электронную почту пользователя.

Эта функция помогает не допускать просрочек по платежам, своевременно выполнять финансовые обязательства и эффективно управлять личным бюджетом и планами.

3.2.4 Финансовые цели и планы

Функция постановки финансовых целей и планов — одна из важных и уникальных возможностей мобильного приложения. Основная цель этой функции — приучить пользователя к регулярному накоплению, формировать финансовую дисциплину и мотивировать его к достижению запланированных целей.

Пользователь может задать любую желаемую цель прямо через приложение. При создании цели указываются:

- Планируемая сумма накоплений;
- Срок достижения;
- Текущая сумма накоплений.

Эта возможность позволяет пользователю сохранить в системе свою реальную финансовую цель и выполнять её поэтапно.

Финансовые цели могут быть различными:

- **Краткосрочные** (выполнение в течение нескольких дней или недель);
- **Среднесрочные** (в течение нескольких месяцев);
- **Долгосрочные** (накопления на несколько лет вперёд).

При добавлении цели пользователь вводит:

- Название цели;
- Планируемую сумму;
- Желаемый срок;
- Текущую накопленную сумму;
- Краткое описание цели.

В приложении процесс выполнения целей отображается с помощью **специальной полосы прогресса**, которая визуально показывает, на сколько процентов цель уже достигнута. Также система автоматически рассчитывает и отображает:

- Сколько средств ещё необходимо накопить;
- Сколько времени осталось до достижения цели, учитывая текущее финансовое состояние пользователя.

Кроме создания цели, предусмотрены функции **редактирования и удаления**, что позволяет пользователю гибко управлять своими финансовыми планами.

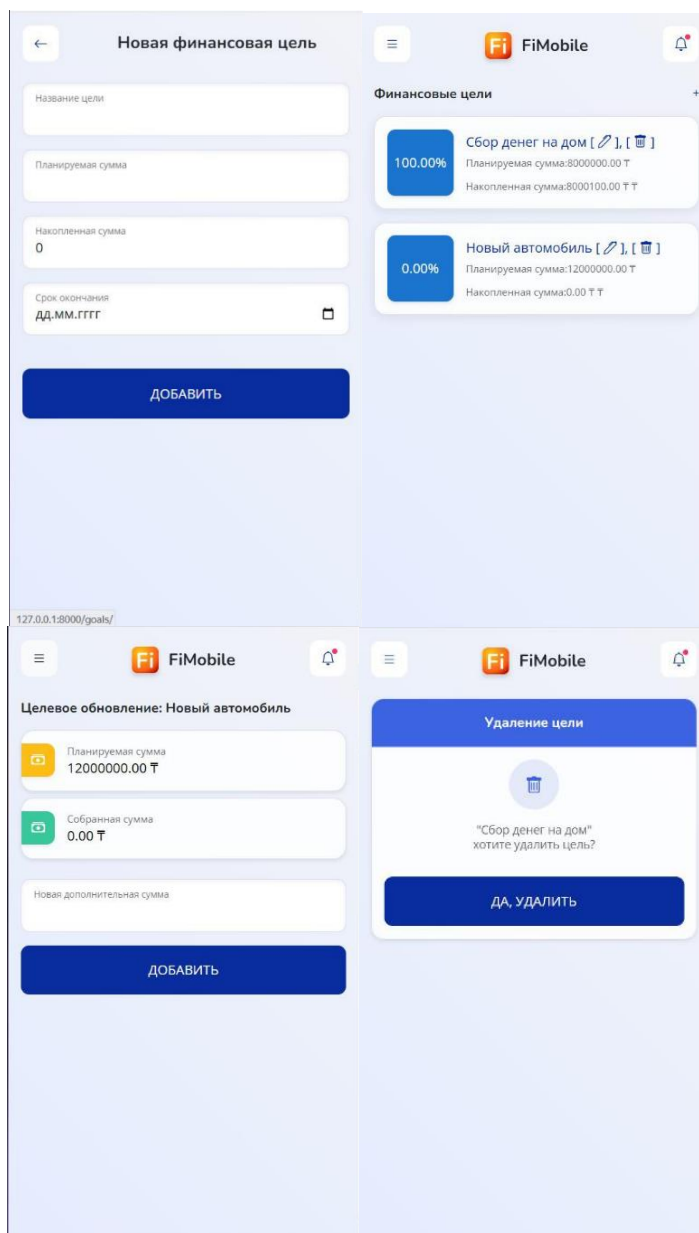


Рисунок 17 – Страница добавления финансовой цели

На рисунке представлена страница добавления финансовой цели. На этой странице пользователь может свободно задать свою цель по накоплению. Структура формы очень простая и понятная. Пользователь заполняет необходимые поля и нажимает кнопку **«Сохранить»**, чтобы добавить цель в систему.

Созданные цели отображаются в отдельном разделе. Для каждой цели указываются:

- её **название**,

- **планируемая сумма,**
- **накопленные средства,**
- **индикатор прогресса.**

Ход выполнения цели отображается в виде **процентного показателя**, что делает визуализацию максимально наглядной и понятной.

Эта функция укрепляет финансовую дисциплину пользователя, помогает планомерно накапливать средства, последовательно достигать поставленных целей и эффективно управлять личными финансами.

В третьей главе были подробно рассмотрены этапы реализации мобильного приложения для управления личными финансами. Описаны основные фазы разработки — от проектирования интерфейса и настройки базы данных до программной реализации ключевых функций, таких как учёт доходов и расходов, аналитика, напоминания и финансовые цели. Все функции были протестированы и интегрированы в единое целостное решение, ориентированное на удобство пользователя, безопасность данных и функциональную полноту. Результатом главы стало создание работающего прототипа, готового к дальнейшему использованию и развитию.

ЗАКЛЮЧЕНИЕ

В дипломной работе всесторонне рассмотрен процесс разработки мобильного приложения для управления личными финансами. В ходе исследования были проанализированы современные методы финансового планирования, теоретические основы управления личными финансами, а также отечественный и зарубежный опыт. Результаты исследования подтвердили актуальность и значимость управления финансовыми операциями с использованием современных информационных технологий.

В первой главе дипломной работы были изучены понятие и значимость управления личными финансами. Было установлено, что эффективное управление финансовыми ресурсами человека включает контроль доходов и расходов, планирование накоплений и инвестиций, а также постановку финансовых целей. Кроме того, рассмотрены современные подходы к повышению уровня финансовой грамотности.

Во второй главе представлен процесс проектирования мобильного приложения для управления личными финансами. Были разработаны архитектура приложения и структура базы данных, определены функциональные требования. Реализованы ключевые модули приложения: учёт доходов и расходов, категоризация трат, постановка финансовых целей, создание аналитических отчётов и диаграмм. Безопасность системы обеспечена на основе фреймворка Django, предприняты все необходимые меры по защите пользовательских данных. В третьей главе описана реализация основных функциональных возможностей приложения и проведено тестирование системы. Результаты тестирования подтвердили полное соответствие приложения заданным требованиям. Интерфейс системы прост и интуитивно понятен, что обеспечивает удобное управление финансовыми операциями для пользователей.

Практическая значимость работы заключается в том, что разработанное мобильное приложение направлено на автоматизацию процессов управления и планирования личных финансов, что способствует обеспечению финансовой стабильности. Использование приложения помогает развивать навыки управления личными финансами, повышает уровень финансовой грамотности и снижает риски. В качестве итогов дипломной работы предлагаются направления дальнейшего развития приложения: улучшение пользовательского интерфейса, внедрение мультязычности, разработка модуля финансового консультанта с использованием технологий искусственного интеллекта, а также интеграция с облачными хранилищами данных. Кроме того, адаптация приложения для мобильных платформ Android и iOS позволит значительно расширить его пользовательскую аудиторию.

В целом, поставленные цели и задачи дипломной работы были полностью выполнены, а полученные результаты и сделанные выводы подтвердили свою актуальность и практическую ценность как в теоретическом, так и в прикладном аспекте.

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

1. Акентьев Р. Финансовая грамотность, или Основы управления личными финансами. – Москва, 2022.
2. Фомина Е. А. Управление личными финансами // Сборник научных работ серии «Финансы, учёт, аудит». – Донецк, 2016. – Вып. 4.
3. Савенок В. С. Личные финансы. Самоучитель. – Санкт-Петербург: Питер, 2017. – 432 с.
4. Чернов А. Ю. Личные финансы. – Москва: Перспектива, 2017. – 176 с.
5. 15 советов о том, как эффективно управлять личными финансами // Investlab – Информационный портал о личных финансах и инвестициях. – Режим доступа: <https://invlab.ru/financy/upravlenie-lichnymi-finansami/> (дата обращения: 21.02.2020).
6. Прусова В. И. Риск-менеджмент в сфере личного финансового планирования как инструмент минимизации последствий кризисных явлений // Экономика и бизнес: теория и практика. – 2022. – № 3-2. – С. 102–106.
7. Колесова О. В., Мустафина Д. Р., Яшкова Е. В. Особенности управления личными финансами в современных условиях // Актуальные вопросы современной экономики. – 2021. – № 2. – С. 384–390.
8. Шерстобитова С. В. Проблема рационализации процесса управления личными финансами // Интернаука. – 2021. – № 38-2 (214). – С. 11–13.
9. Савельева Е. Россияне стали активнее пользоваться сервисами для управления личными финансами: исследование. – Москва, 2022. – Режим доступа: <https://www.banki.ru/news/lenta/?id=10959817> (дата обращения: 25.10.2023).
10. Авдюшина М. А., Агеева Е. В., Андреев П. А. и др. Финансово-правовая грамотность экономических субъектов в условиях цифровизации экономики. – Иркутск: Байкальский государственный университет, 2022. – 294 с.
11. Сорокина С. В. Современные подходы к управлению личными финансами в контексте повышения финансовой грамотности // Актуальные вопросы учета и управления в условиях информационной экономики. – 2022. – № 4. – С. 419–423.
12. Родина Л. А., Завьялова Л. В. Управление личными финансами в современных условиях // Вестник ОмГУ. Серия: Экономика. – 2020. – № 4. – С. 36–47.
13. Лусникова, Е. А. (2019). Управление личными финансами: современные подходы и инструменты. *Финансовая аналитика: проблемы и решения*, 12(3), 45–52.
14. Кузнецова, И. В. (2020). Цифровизация управления личными финансами в условиях развития финансовых технологий. *Экономика и предпринимательство*, 11(1), 78–84.
15. Смирнова, Н. А. (2021). Роль мобильных приложений в управлении личными финансами. *Журнал экономических исследований*, 5(2), 112–118.

16. Goyal, K., & Kumar, S. (2021). Antecedents and consequences of personal financial management behavior: A systematic literature review and future research agenda. *International Journal of Bank Marketing*, 39(7), 1166–1207. [scholar.google.com+1scholar.google.com+1](#)
17. Walsh, B., & Lim, H. N. (2020). Millennials' adoption of personal financial management (PFM) technology and financial behavior. *Financial Planning Review*, 3(3), e1095. [scholar.google.com](#)
18. Dew, J. P., & Xiao, J. J. (2011). The financial management behavior scale: Development and validation. *Journal of Financial Counseling and Planning*, 22(1), 43–59. [scholar.google.com+4scholar.google.com+4scholar.google.com+4](#)
19. Finke, M. S., Huston, S. J., & Winchester, D. D. (2011). Financial advice: Who pays. *Journal of Financial Counseling and Planning*, 22(1), 18–26. [scholar.google.com](#)
20. Olafsson, A., & Pagel, M. (2018). The liquid hand-to-mouth: Evidence from personal finance management software. *The Review of Financial Studies*, 31(3), 1099–1136. [scholar.google.com+2scholar.google.com+2scholar.google.com+2](#)
21. Pearson, B. M. (2022). The role of personal financial salience: Part II. *Journal of Financial Planning*, 35(9), 60–69. [scholar.google.com](#)
22. Liu, Z., & James, R. (2020). Finding the next major donor: The relationship between financial planning horizon and charitable giving. *Journal of Personal Finance*, 19(1), 10–25. [scholar.google.com](#)
23. Goetz, J. W., Palmer, L., Zhang, L., & Chatterjee, S. (2020). Changes in household net financial assets after the Great Recession: Did financial planners make a difference? *Journal of Financial Counseling and Planning*, 31(1), 45–59. [arXiv](#)
24. Frees, D., Gangal, A., & Shaviro, C. (2024). Quantifying the causal effect of financial literacy courses on financial health. *arXiv preprint arXiv:2405.01789*. [arXiv](#)
25. Godbole, A., Shah, Z., & Mudholkar, R. S. (2025). Preventing household bankruptcy: The one-third rule in financial planning with mathematical validation and game-theoretic insights. *arXiv preprint arXiv:2501.15557*.
26. Сахетгельдиев А. Финансовые приложения и инструменты для управления деньгами // *Ceteris Paribus*. – 2024. – №10. – С. 45–52. [КиберЛенинка](#)
27. Лобов А. В., Мамонов К. А., Чернышева Т. Ю. Мобильное приложение для контроля расходов и доходов // *Материалы международной научно-практической конференции*. – 2022. – С. 176–183. [Электронная библиотека ТюмГУ](#)
28. Яловчук В. В. Распределённая система для управления финансовыми ресурсами: автореферат диссертации на соискание степени магистра. – Минск: БГУИР, 2020. [Libeldoc](#)
29. Есюнин Н. О., Пантимионов М. В. Разработка требований к мобильному приложению по управлению личными финансами // *Студенческий научный форум*. – 2013. [Научный форум](#)
30. Шожат А. А. Разработка мобильного приложения на Android для управления личными финансами: выпускная квалификационная работа бакалавра. – Санкт-Петербург: СПбПУ, 2024. [Электронная библиотека СПбПУ](#)

31. Родина Л. А., Завьялова Л. В. Управление личными финансами в современных условиях // Вестник ОмГУ. Серия: Экономика. – 2020. – №4. – С. 36–47.
32. García-Méndez S., Fernández-Gavilanes M., Juncal-Martínez J., González-Castaño F. J., Barba Seara O. Identifying Banking Transaction Descriptions via Support Vector Machine Short-Text Classification Based on a Specialized Labelled Corpus // arXiv preprint arXiv:2404.08664. – 2024.[arXiv](https://arxiv.org/abs/2404.08664)
33. Digit: AI-powered personal finance assistant. – Доступно на: <https://digit.co>
34. Kaushik P. Is Artificial Intelligence the Way Forward for Personal Finance? // WIRED. – Доступно на: <https://www.wired.com/story/is-artificial-intelligence-the-way-forward-for-personal-finance/>[Википедия — свободная энциклопедия](#)
35. Aljazeera. Can AI Predict Stock Market Movements? – Доступно на: <https://www.aljazeera.com/program/inside-story/2019/12/12/can-ai-predict-stock-market-movements>
36. Beyond Robo-Advisers: How AI Could Rewire Wealth Management // American Banker. – Доступно на: <https://www.americanbanker.com/news/beyond-robo-advisers-how-ai-could-rewire-wealth-management>[Википедия — свободная энциклопедия](#)
37. ZestFinance Introduces Machine Learning Platform to Underwrite Millennials and Other Consumers with Limited Credit History // Business Wire. – Доступно на: <https://www.businesswire.com/news/home/20170627005397/en/ZestFinance-Introduces-Machine-Learning-Platform-to-Underwrite-Millennials-and-Other-Consumers-with-Limited-Credit-History>[Википедия — свободная энциклопедия](#)
38. FBI Warns of Growing Use of Generative AI in Financial Fraud Schemes // CISOCLUB. – 2024-12-08. – Доступно на: <https://cisoclub.ru/fbi-warns-of-growing-use-of-generative-ai-in-financial-fraud-schemes/>