**1 Анализ задачи**

**1.1 Описание предметной области**

В современном мире финансовый учет стал неотъемлемой частью деятельности любого предприятия. Многие организации используют специализированные системы для управления финансами, ведения бухгалтерского учета и формирования отчетности. В связи с этим разработка веб-приложений для финансового учета стала не только важным, но и перспективным направлением развития информационных технологий.

Финансовые веб-приложения, как часть корпоративных информационных систем, предоставляют пользователям доступ к широкому спектру финансовых данных и операций, а также позволяют эффективно управлять финансовыми потоками. Поэтому создание качественных и функциональных финансовых систем является важной задачей для большинства компаний, предпринимателей и финансовых учреждений.

Финансовое веб-приложение представляет собой совокупность связанных между собой модулей, обеспечивающих ведение финансового учета, формирование отчетности и анализ финансовых показателей. В зависимости от целей и задач, которые ставит перед собой организация, система может иметь различные виды и функциональность.

Одним из важных аспектов разработки современных финансовых систем является безопасность, то есть способность защищать конфиденциальные финансовые данные от несанкционированного доступа. Система безопасности позволяет контролировать доступ к финансовой информации, обеспечивая защиту данных и аудит всех операций.

Еще одним важным аспектом разработки финансовых систем является надежность, которая означает возможность корректной работы системы при обработке большого объема финансовых данных. Системы должны быть доступными и удобными для использования как на стационарных компьютерах, так и на мобильных устройствах.

В зависимости от целей и задач, финансовые системы могут быть разных типов: системы бухгалтерского учета, системы управления финансами, системы финансового планирования, системы финансового анализа и другие. Каждый тип системы имеет особенности и требует соответствующего подхода к разработке и внедрению.

Пользователи системы могут не только вести финансовый учет, но и получать аналитическую информацию о финансовом состоянии организации, принимать участие в процессе бюджетирования и контролировать исполнение бюджета.

**1.2 Обзор существующих аналогов**

Mint Функционал: Mint помогает пользователям отслеживать доходы, расходы, составлять бюджеты и управлять счетами. Приложение автоматически синхронизируется с банковскими счетами, кредитными картами и инвестициями, предоставляя подробную аналитику финансов.

Особенности:

Автоматическая категоризация расходов;

Уведомления о предстоящих платежах;

Анализ кредитного рейтинга.

Целевая аудитория: Индивидуальные пользователи, желающие улучшить свои навыки управления личными финансами.

Аналог "Mint" представлен на рисунке 2.1.

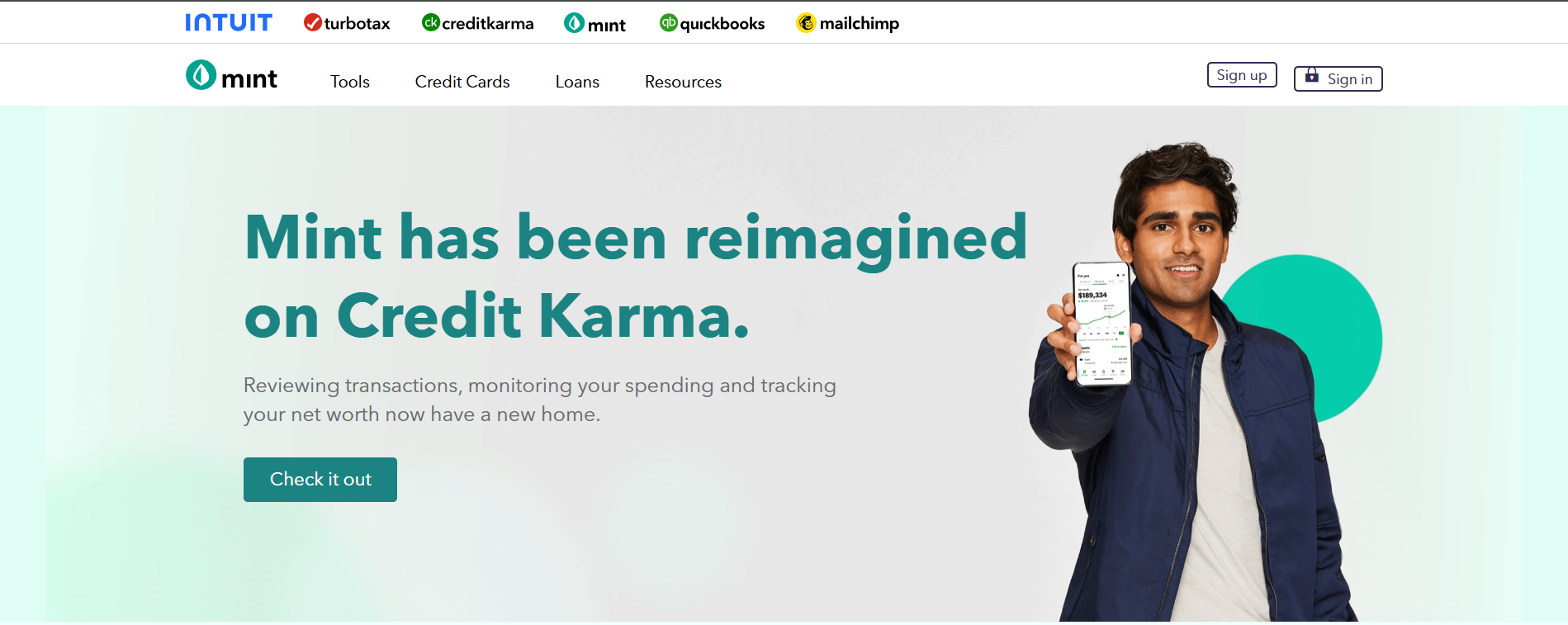


Рисунок 2.1 – Сервис "Mint"

Источник: https://mint.intuit.com/

Достоинства:

Удобный и яркий интерфейс, привлекающий внимание.

Автоматизация: Приложение автоматически синхронизируется с банковскими счетами и кредитными картами, минимизируя ручной ввод данных.

Удобство: Простая и интуитивно понятная навигация, подходящая для новичков.

Информативность: Предоставляет детализированные отчёты о расходах, доходах и инвестициях.

Недостатки:

Ограниченный контроль: Автоматическая категоризация может быть неточной, что требует ручной корректировки.

Реклама: Бесплатная версия включает рекламу, что может раздражать пользователей. Локализация: Поддержка некоторых функций ограничена для пользователей за пределами США.

YNAB (You Need A Budget) Функционал: YNAB ориентировано на детальное планирование бюджета и помогает пользователям распределять каждую заработанную единицу дохода по категориям. Оно направлено на формирование правильных финансовых привычек.

Особенности:

Принцип "каждый доллар имеет цель";

Синхронизация с банковскими счетами в реальном времени;

Подробные отчёты о расходах и прогресс в достижении финансовых целей.

Целевая аудитория: Люди, стремящиеся к строгому финансовому контролю и долгосрочному планированию.

Аналог " YNAB " представлен на рисунке 2.2.

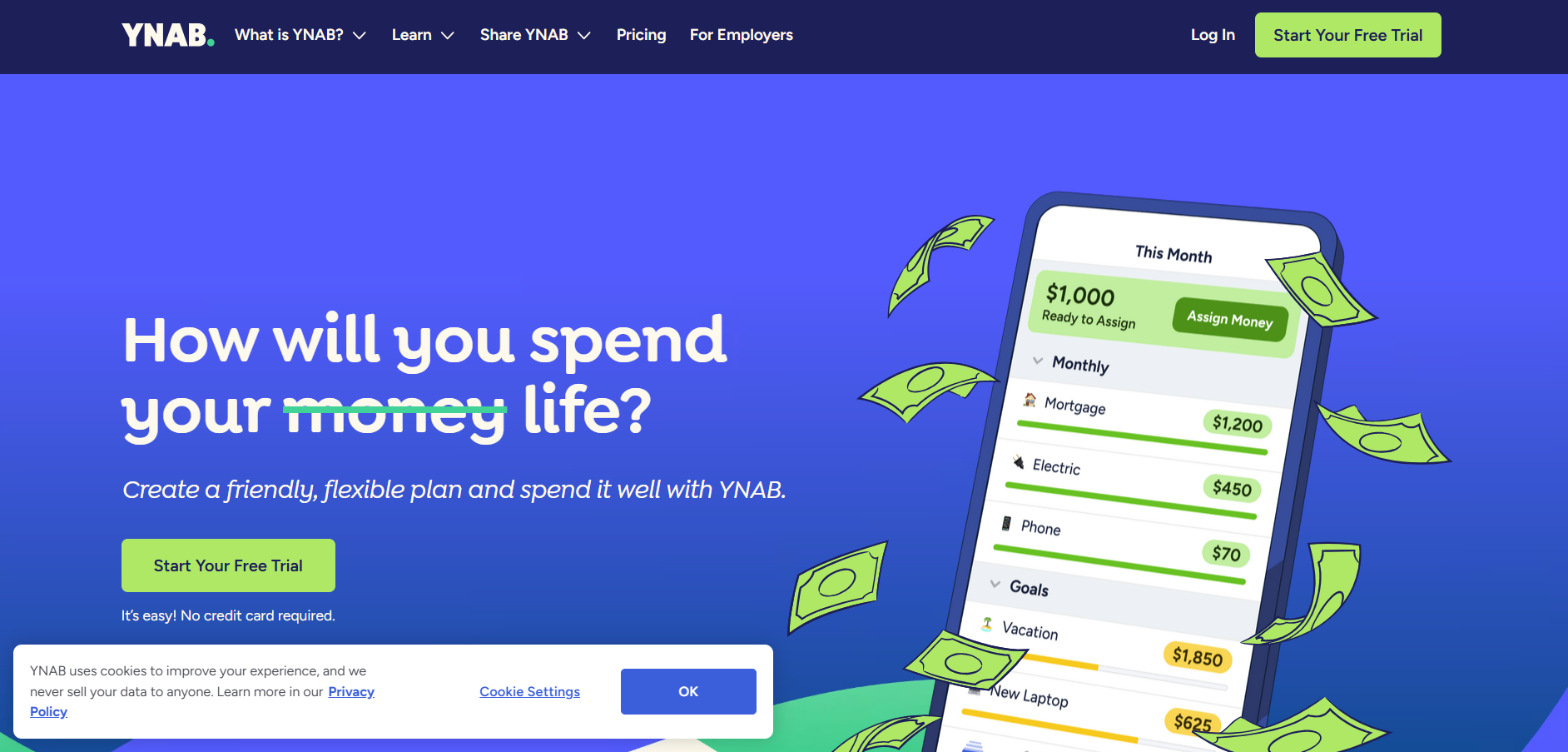


Рисунок 2.2 - Сервис " YNAB "

Источник: https://www.ynab.com/

Достоинства:

Фокус на планировании: Помогает пользователям установить финансовые цели и управлять бюджетом, основываясь на принципе "каждый доллар имеет цель".

Обучение: Предоставляет обучающие материалы и курсы для улучшения финансовой грамотности.

Прозрачность: Рассчитано на тех, кто хочет детально отслеживать свои расходы и планировать сбережения.

Недостатки:

Платность: Приложение требует подписки, что может быть недостатком для пользователей с ограниченным бюджетом.

Сложность: Для новичков интерфейс и подход к планированию бюджета могут показаться сложными.

Отсутствие автоматизации: В отличие от других приложений, YNAB требует больше ручного ввода данных, что может занимать время.

PocketGuard Функционал: PocketGuard позволяет пользователям отслеживать расходы, предоставляя информацию о том, сколько денег доступно для траты после учёта обязательных расходов и сбережений.

Особенности:

Автоматический расчет доступного остатка ("сколько можно потратить");

Подключение банковских счетов;

Анализ подписок и рекомендация по их оптимизации.

Целевая аудитория: Люди, ищущие простое и понятное решение для контроля повседневных расходов.

Третий аналог " PocketGuard " представлен на рисунке 2.3.

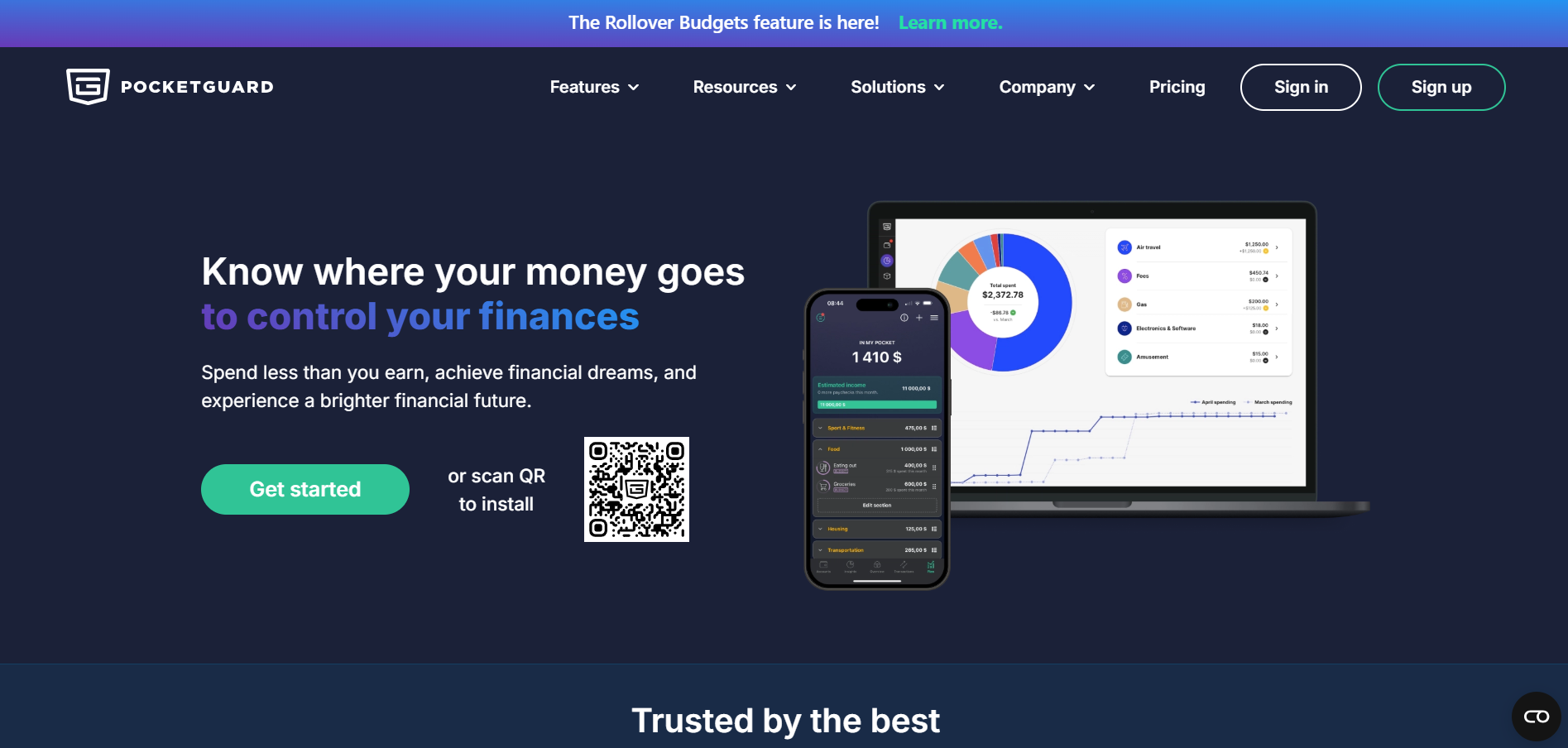


Рисунок 2.3 – Сервис " PocketGuard "

Источник: https://pocketguard.com/

Достоинства:

"Доступно для траты": Рассчитывает, сколько денег можно потратить после учёта фиксированных расходов и сбережений.

Лёгкость использования: Простой интерфейс, который удобен для пользователей с минимальным опытом управления финансами.

Оптимизация подписок: Помогает выявить ненужные подписки и предлагает их отменить для экономии средств.

Недостатки:

Ограниченный функционал: Приложение больше ориентировано на повседневные расходы, чем на долгосрочное финансовое планирование.

Платный премиум-доступ: Бесплатная версия имеет ограниченные функции, такие как отсутствие детализированных бюджетных планов.

Точность: Автоматическая категоризация расходов может быть неточной, особенно для нестандартных транзакций.

Глобальная ограниченность: Некоторые функции недоступны за пределами стран, поддерживаемых приложением.

**1.3 Постановка задачи. Основные требования к программному продукту**

В соответствии с разграничением прав доступа каждой роли предоставляется определённый список требований.

Для пользователей системы реализуются следующие функции: авторизация, регистрация, вход, просмотр финансовых отчетов, фильтрация и поиск финансовых операций, добавление часто используемых отчетов в избранное, создание комментариев к финансовым операциям, оценка эффективности финансовых операций, просмотр графиков и диаграмм финансовых показателей.

Серверная часть включает базу данных финансовых операций с поддержкой обработки SQL-запросов, поиск по фрагменту текста в финансовых документах, систему уведомлений о важных финансовых событиях.

Для администратора системы доступны следующие функции: добавление, редактирование финансовых операций, управление пользователями и правами доступа, настройка системы безопасности, управление шаблонами отчетов.

К основным требованиям к разрабатываемой системе относятся: профессиональный и понятный интерфейс, профиль пользователя с возможностью редактирования персональной информации и просмотра истории операций, система аналитики — предложение пользователям финансовых рекомендаций, основанное на анализе предыдущих операций и финансовых показателей, оптимизация под различные конфигурации оборудования, корректное и бесперебойное функционирование, необходимость обеспечения высокой производительности, стабильность работы веб-приложения, быстрая загрузка страниц и быстрый поиск финансовой информации.

Средой разработки выбрана IntelliJ IDEA и VS Code благодаря встроенной поддержке Java, Spring Boot, HTML и CSS, удобной системе отладки, интеграции с Git и богатому набору расширений. Visual Studio Code предоставляет функции автодополнения кода, подсветки синтаксиса и встроенный терминал для работы с командной строкой.

Для управления зависимостями и пакетами используется Maven - менеджер пакетов для Java. Maven позволяет легко устанавливать и обновлять необходимые библиотеки, такие как Spring Framework и различные компоненты для работы с безопасностью.

В процессе разработки активно применяются инструменты для работы с базой данных - MySQL Workbench, позволяющий удобно просматривать и редактировать структуру базы данных, выполнять SQL-запросы и управлять пользователями. Для контроля версий используется система Git, обеспечивающая эффективную командную работу и отслеживание изменений в коде.

Для создания пользовательского интерфейса применяется фреймворк Thymeleaf, предоставляющий готовые компоненты и стили для создания адаптивного и современного дизайна. Thymeleaf значительно ускоряет разработку frontend-части приложения и обеспечивает корректное отображение на различных устройствах.