

Продукт: Мобильное приложение для управления финансами

Границы Продукта:

1. Пользовательская аутентификация и регистрация.
2. Возможность добавления и управления банковскими счетами и кредитными картами.
3. Отслеживание доходов и расходов с возможностью автоматической категоризации.
4. Создание и управление бюджетами и целями.
5. Графики и отчеты по финансовому состоянию.
6. Возможность установки оповещений о платежах и близких к истечению сроках.
7. Интеграция с банковскими API для автоматической синхронизации данных.
8. Защита данных и конфиденциальность пользователей.

Критерии завершённого Продукта:

1. Возможность регистрации и аутентификации пользователей.
2. Добавление и управление финансовыми аккаунтами.
3. Отслеживание транзакций, включая доходы и расходы.
4. Создание бюджетов и целей, а также отслеживание их выполнения.
5. Генерация графиков и отчетов по финансам.
6. Оповещения о предстоящих платежах.
7. Интеграция с хотя бы двумя банками через их API.
8. Механизм шифрования и обеспечения конфиденциальности данных.

Команда Разработки:

1. Продуктовый менеджер - ответственный за определение функциональных требований и приоритетов разработки.
2. UI/UX дизайнер - создание пользовательского интерфейса приложения.
3. Разработчик (Frontend) - создание клиентской части мобильного приложения (iOS и Android).
4. Разработчик (Backend) - создание серверной части приложения и интеграция с банковскими API.
5. Тестировщик - проверка функциональности и безопасности приложения.
6. DevOps инженер - обеспечение непрерывной интеграции и развертывания приложения.
7. Специалист по информационной безопасности - обеспечение защиты данных и конфиденциальности.

Матрица компетенций для Команды Разработки:

Роль	Ключевые компетенции
Продуктовый менеджер	- Анализ рынка и потребительских потребностей - Определение функциональных требований - Управление приоритетами и планирование разработки
UI/UX дизайнер	- Коммуникация с заказчиками и пользователями - Дизайн пользовательского интерфейса - Прототипирование и тестирование интерфейса

Роль	Ключевые компетенции
Разработчик (Frontend)	<ul style="list-style-type: none"> <li>- Визуальное оформление и работа с цветовой палитрой</li> <li>- Знание языков программирования (например, Swift, Kotlin)</li> <li>- Разработка мобильных приложений</li> <li>- Работа с пользовательскими интерфейсами</li> <li>- Оптимизация производительности приложения</li> </ul>
Разработчик (Backend)	<ul style="list-style-type: none"> <li>- Знание языков программирования (например, Python, Java)</li> <li>- Создание и управление базой данных</li> <li>- Работа с API банков и сторонними сервисами</li> <li>- Обеспечение безопасности данных</li> </ul>
Тестировщик	<ul style="list-style-type: none"> <li>- Тестирование функциональности приложения</li> <li>- Создание тестовых сценариев и тест-кейсов</li> <li>- Обнаружение и отслеживание ошибок</li> </ul>
DevOps инженер	<ul style="list-style-type: none"> <li>- Настройка системы непрерывной интеграции и развертывания</li> <li>- Управление серверной инфраструктурой</li> <li>- Автоматизация процессов разработки</li> </ul>
Специалист по информационной безопасности	<ul style="list-style-type: none"> <li>- Защита данных и обеспечение конфиденциальности</li> <li>- Мониторинг уязвимостей и рисков</li> <li>- Разработка политики безопасности и ее соблюдение</li> </ul>

Команда разработки должна тесно сотрудничать, обмениваться информацией и обеспечивать высокое качество и безопасность продукта на каждом этапе разработки.

Недавно я изучил описание неудачного опыта внедрения Scrum в одной организации и могу предоставить тезисы причин, которые привели к неудаче:

1. Недостаточное обучение и понимание: Команда и руководство организации не были достаточно обучены и не поняли полностью, что такое Scrum и как он работает. Они начали применять Scrum, не имея ясного представления о его основных принципах и целях.
2. Недостаточная поддержка руководства: Руководство организации не поддерживало полностью процесс внедрения Scrum. Они ожидали быстрых результатов и продолжали вмешиваться в работу команды, не давая ей достаточной автономии.
3. Отсутствие культуры обратной связи: В организации отсутствовала культура обратной связи, что привело к тому, что проблемы и барьеры внедрения Scrum не решались своевременно. Команда не могла выразить свои замечания и предложения.

4. Неадекватные ожидания: Руководство ожидало, что Scrum решит все проблемы и приведет к мгновенным улучшениям в производительности. Но в реальности Scrum - это изменение культуры и процесса, которое требует времени и усилий.
5. Отсутствие выделенного Scrum-мастера: В организации не было назначенного Scrum-мастера, который был бы ответственен за обучение и сопровождение команды в процессе внедрения. Это привело к тому, что команда оставалась без руководства.
6. Инерция и сопротивление изменениям: Сотрудники организации привыкли к старым способам работы и сопротивлялись изменениям, которые вносил Scrum. Это создавало конфликты и замедляло процесс внедрения.
7. Неясные роли и обязанности: Роли внутри команды не были четко определены, что приводило к путанице и дублированию работ. Команда не понимала, кто за что отвечает.
8. Недостаточное тестирование и адаптация: Организация не проводила достаточное тестирование процесса Scrum и не адаптировалась к выявленным проблемам. Вместо этого, они продолжали настаивать на неэффективных подходах.

Эти тезисы отражают основные причины неудачи внедрения Scrum в данном случае. Внедрение Scrum требует серьезных усилий, обучения и понимания, а также поддержки со стороны руководства и готовности к изменениям в культуре и процессах работы.