

Раздаточный материал № 1



Раздаточный материал № 2 (самостоятельно)

Вспомогательные процессы

Процесс документирования. В процессе разработки и далее исполнитель пишет документацию и руководства пользователя к разрабатываемому программному продукту. Данные документы помогут разработчикам [вспомнить/разобраться] структуру и код ПО (ибо со временем всё забывается, особенно в больших проектах), а пользователям освоить работу с программой.

Процесс управления конфигурацией. Данный процесс включается в себя работы по управлению наборами разрабатываемых компонентов ПО и по управлению версиями ППП.

Процесс обеспечения качества. Он отвечает за то, чтобы разрабатываемый программный продукт соответствовал предварительным требованиям к разработке, а также стандартам организаций исполнителя и заказчика.

Процесс верификации. Нужен для того, чтобы выявить ошибки, внесённые в ПО во время конструирования, а также выявить несоответствия разрабатываемого ПО выработанной архитектуре.

Процесс аттестации. Процесс направлен на подтверждение соответствия получаемых величин эталонным. То есть, выходные данные должны иметь погрешность, удовлетворяющую требованиям и установленным стандартам.

Процесс совместной оценки. Нужен для контроля и проверки состояния персонала и разрабатываемого программного продукта. Выполняется обеими сторонами (заказчиком и исполнителем) на протяжении времени всех работ по проекту.

Процесс аудита. Аудит направлен на независимую оценку текущих положений, состояния проекта, документации и отчетов. При аудите выполняется сравнение с договором и документами, определяющими стандарты. Может выполняться также обеими сторонами.

Процесс разрешения проблем. Реализует устранение недочётов, выявленных во время всех процессов, связанных с контролем и оценкой.

Организационные процессы жизненного цикла программного продукта

Существует и проводится ряд мер, направленных на повышение организации и качества разработки программного обеспечения. Они называются организационными процессами жизненного цикла.

Процесс управления, который направлен на грамотное и эффективное управлением персонал компании-исполнителя. За это отвечают люди, находящиеся на руководящих постах, а также специальный отдел в фирме.

Процесс создания инфраструктуры. Разработка программных продуктов требует наличия огромного количества инфраструктурных компонентов: компьютеров, серверов, специальных программ для разработки и т.д. Кроме того, готовый продукт требует наличия определённых единиц для его работы. Данный процесс необходим для подготовки оборудования и ПО для разработчиков, а также для успешного функционирования готового ПП у заказчика.

Процесс усовершенствования. Направлен на усовершенствование всех остальных процессов жизненного цикла программного обеспечения. Усовершенствование может повысить производительность разработчиков и добиться большей выгоды от выполнения заказа на производство программы.

Процесс обучения. Постоянное обучение сотрудников и повышение их квалификации — это залог производства качественных продуктов и программ. Процесс обучения направлен на организацию мероприятий для повышения уровня и получения новых навыков сотрудниками компании-разработчика.

Раздаточный материал № 3

Software Development Life Cycle (SDLC) – жизненный цикл программного обеспечения



Раздаточный материал № 4



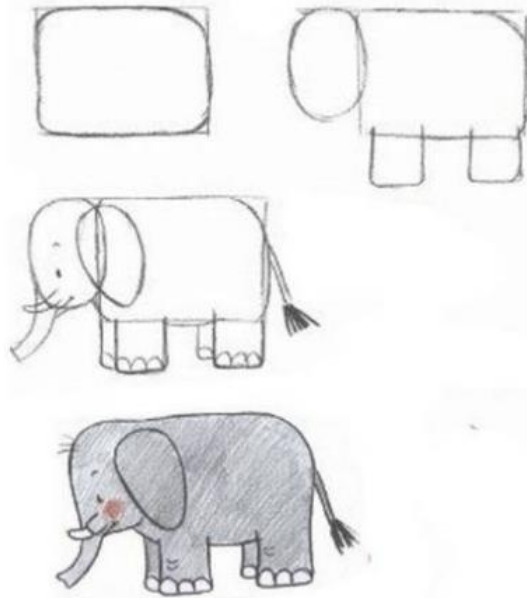
Раздаточный материал № 5



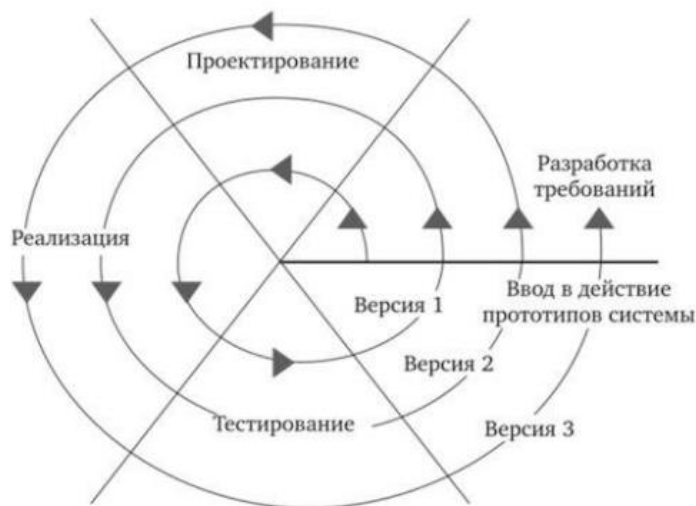
Раздаточный материал № 6



Раздаточный материал № 7



Раздаточный материал № 8



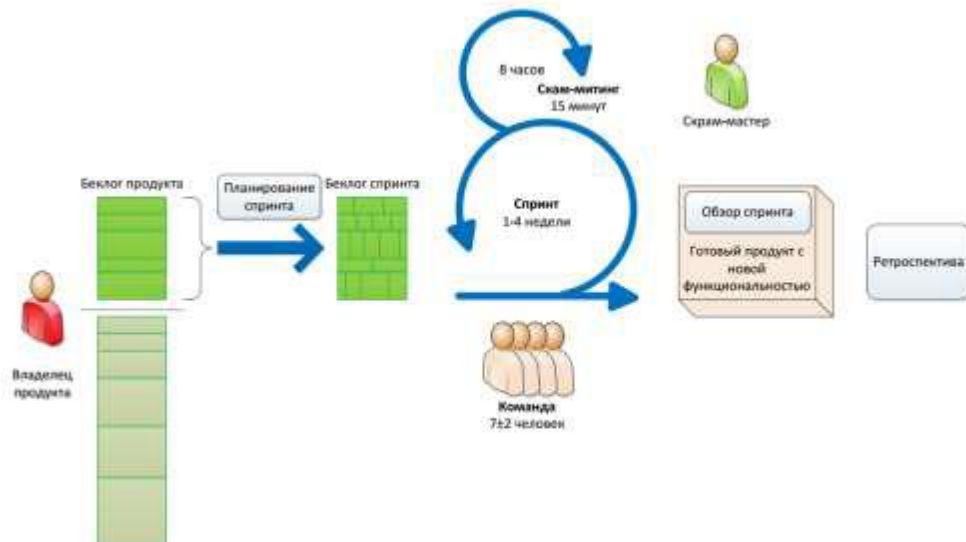
Раздаточный материал № 9 (самостоятельно)

Основополагающие принципы Agile Manifesto:

- наивысшим приоритетом признается удовлетворение заказчика за счёт ранней и бесперебойной поставки ценного программного обеспечения;
- изменение требований приветствуется даже в конце разработки (это может повысить конкурентоспособность полученного продукта);
- частая поставка работающего программного обеспечения (каждые пару недель или пару месяцев с предпочтением меньшего периода);
- общение представителей бизнеса с разработчиками должно быть ежедневным на протяжении всего проекта;
- проекты следует строить вокруг заинтересованных людей, которых следует обеспечить нужными условиями работы, поддержкой и доверием;
- самый эффективный метод обмена информацией в команде — личная встреча;

работающее программное обеспечение — лучший измеритель прогресса;
спонсоры, разработчики и пользователи должны иметь возможность поддерживать постоянный темп на неопределённый срок;
постоянное внимание к техническому совершенству и хорошему проектированию увеличивают гибкость;
простота, как искусство не делать лишней работы, очень важна;
лучшие требования, архитектура и проектные решения получаются у самоорганизующихся команд;
команда регулярно обдумывает способы повышения своей эффективности и соответственно корректирует рабочий процесс.

Раздаточный материал № 10



Раздаточный материал № 11

Роли:

- Scrum мастер (Scrum Master),
- Владелец продукта (Product Owner),
- Команда (Team).

Раздаточный материал № 12

Артефакты:

- Product Backlog (Бэклог Продукта),
- Sprint Backlog (Бэклог Спринта),
- Increment (Инкремент).

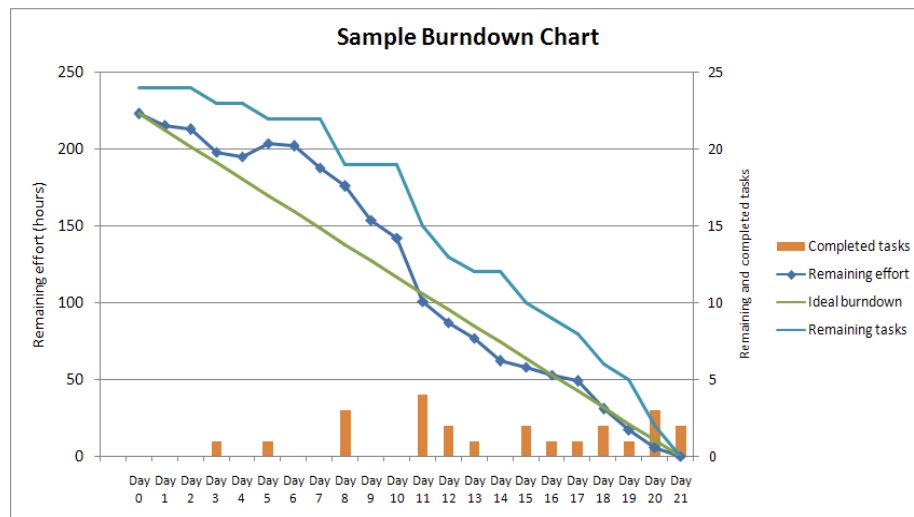
Раздаточный материал № 13

f_x	A	B
1	Приоритет	Компонент
2	100	Каталог товаров
3	99	Корзина заказа
4	80	Оплата по Робокассе
5	70	Форма расчета стоимости доставки
6	65	Фильтр товаров
7	60	Форма обратного звонка
8	50	Личный кабинет
9	40	Поиск как на эппл.ком :)
10	20	Система отзывов
11	16	Анимированное добавление товаров в корзину
12	15	3d-крутилка товаров
13	10	Социальная сеть

Раздаточный материал № 14



Раздаточный материал № 15



Раздаточный материал № 16

Проектирование

- Проведение интервью с заказчиком
- Определение требований к программному продукту
- Составление ТЗ
- Построение диаграмм (диаграмма вариантов использования, карта сайта)
- Выбор программных и инструментальных средств разработки
- + Добавить карточку

Разработка серверной части веб-приложения

- Построение диаграммы базы данных
- Разработка базы данных при помощи ORM Sequelize
- Создание полноценного сервера для REST API приложения
- Объединение сервера и базы данных при помощи фреймворка Express
- Обеспечение взаимодействия серверной и клиентской частей
- Описание документации сервера при помощи Docker
- + Добавить карточку

Тестирование

- Формирование плана тестирования
- Формирование тест-кейсов
- Проведение тестирования
- Исправление найденных ошибок
- + Добавить карточку

Разработка документации

- Формирование структуры разрабатываемой документации
- Создание системы помощи в веб-приложении
- + Добавить карточку

Beth's Projects

To Do

- Aug 4
- August Newsletter
- Aug 7
- Aug 11
- Webinar #1
- Aug 14
- T21 #1
- Aug 15
- Elem Literacy Workshop
- Aug 18

Doing

- Leveraging - Cambridge
- Jul 24
- Writing - Cambridge
- Jul 21
- Aug 1 T21 #1
- 2/3
- Jul 8
- Eventbrite for iPad Summit Boston
- 1/4
- BB
- ETT Summit
- 0/5
- Jul 28
- Conference Proposals
- 3/5
- Cue Weekly Tweets
- Jul 21

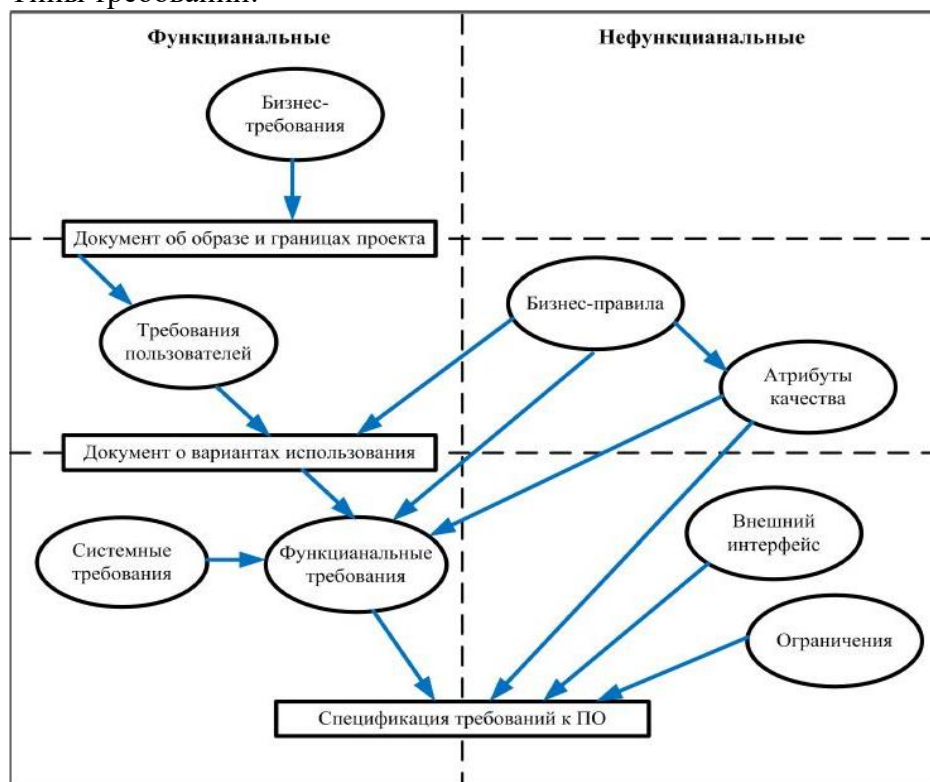
Done

- Elementary - Chicago
- Jun 23
- iPads in Elem - Cambridge
- Jul 14
- Elementary - ATL
- Jun 12
- Cue Weekly Tweets
- 3/4
- Jul 14
- Cue Weekly Tweets
- 4/8
- Jun 16
- Cue Weekly Tweets
- 3/3
- Jul 7
- July Newsletter
- Jul 8



Раздаточный материал № 17

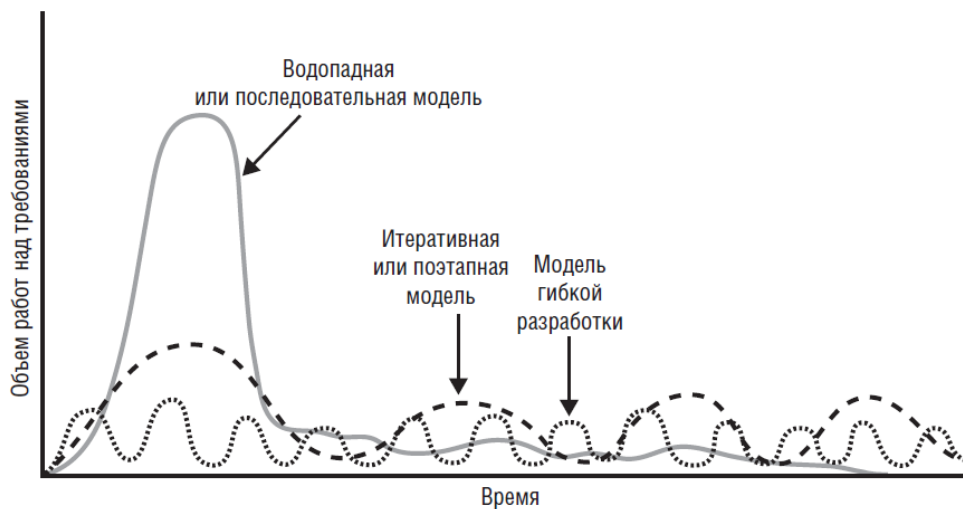
Типы требований:



Раздаточный материал № 18



Раздаточный материал № 19



Раздаточный материал № 20 (самостоятельно)

Выбор сторонника продукта (product champion) в каждом классе пользователей. Это человек, который сможет точно передавать настроения и нужды клиентов. Он представляет потребности определенного класса пользователей и принимает решения от их лица.

Проведение фокус-групп типичных пользователей. Создайте группы типичных пользователей предыдущих версий вашего продукта или похожих. Выясните у них подробности функциональности и качественных характеристик разрабатываемого продукта. Фокус-группы особенно ценны при разработке коммерческих продуктов, когда приходится иметь дело с большой и разнородной клиентской базой. В отличие от сторонников продукта, у фокус-групп обычно нет полномочий на принятие решений.

Проведение совместных семинаров. Совместные семинары по выявлению требований, где тесно сотрудничают аналитики и клиенты — отличный способ выявить нужды пользователей и составить наброски документов с требованиями.

Раздача опросных листов. Это один из способов определения потребностей больших групп пользователей. Опросные листы удобны при работе с любыми большими группами пользователей, но особенно полезны в распределенных группах. Качественные вопросы позволяют быстро выявить аналитическую информацию о потребностях. На основе результатов опросных листов можно более целенаправленно применять дополнительные усилия.

Анализ документов. Имеющаяся документация может помочь выявить, как системы работают сейчас или что они должны делать. К документации относится вся письменная информация о текущих системах и бизнес-процессах, спецификации требований, исследования конкурентов и руководства имеющихся коммерческих программных пакетов. Просмотр и анализ этих документов может помочь выявить функциональность, которая должна остаться и которая больше не нужна, а также определить, как сейчас люди выполняют свою работу, что предлагают конкуренты и что говорят поставщики о том, что должно делать их ПО.

Изучение отчетов о проблемах работающих систем с целью поиска новых идей. Поступающие от клиентов отчеты о проблемах и предложения о расширении функциональности — отличный источник идей о возможностях, которые можно реализовать в следующей версии или новом продукте. За подобной информацией стоит обратиться к персоналу службы поддержки.

Повторное использование требований. Если необходимая клиенту функциональность аналогична уже реализованной в другом продукте, подумайте, готовы ли клиенты гибко пересмотреть свои требования для использования существующих компонентов. Требования, соответствующие бизнес-правилам компании, можно применить в нескольких проектах, например, требования к безопасности, глоссарии, модели и определения данных, профили заинтересованных лиц, описаний классов и архетипы пользователей.

Раздаточный материал № 21

<Роль> должен иметь возможность <возможность> в <показатель производительности> с <момент отсчета> в <условия эксплуатации>, чтобы <ценность>

Например, Администратор клиники должен иметь возможность просмотреть данные о прошлых и запланированных посещениях пациента в течение 3 секунд после определения личности клиента по номеру телефона входящего звонка, чтобы добавить новое или изменить запланированное посещение.

<Система> должна <выполняемая функция> <объект> каждые <производительность> <единица измерения>, чтобы <ценность>.

Например, CRM-система должна отправлять СМС-напоминание клиенту о предстоящем посещении за сутки перед посещением, чтобы он помнил о приеме и пришел вовремя.

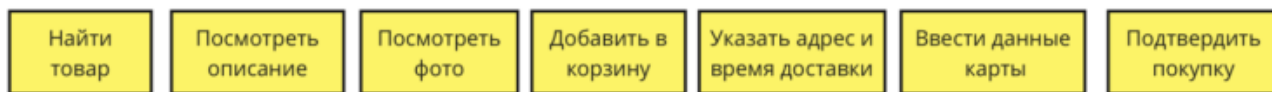
Как администратор базы данных, я хочу автоматически объединять наборы данных из разных источников, чтобы мне было проще создавать отчеты для моих внутренних клиентов.

Как руководитель удаленной группы, я хочу, чтобы в наше приложение для обмена сообщениями для команды было включено совместное использование файлов и аннотации, чтобы моя команда могла сотрудничать в режиме реального времени и хранить архив своей работы в одном месте.

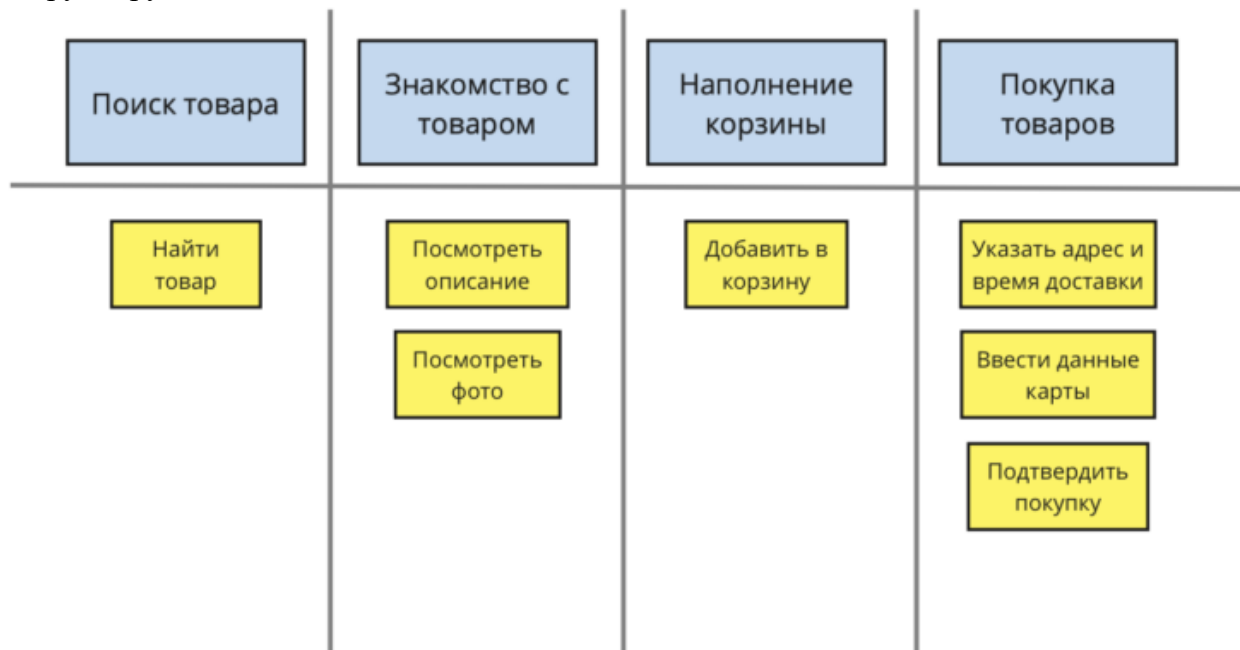
Раздаточный материал № 22

Магазин цветов решил запустить сайт. Визуализируем опыт клиентов с помощью техники USM.

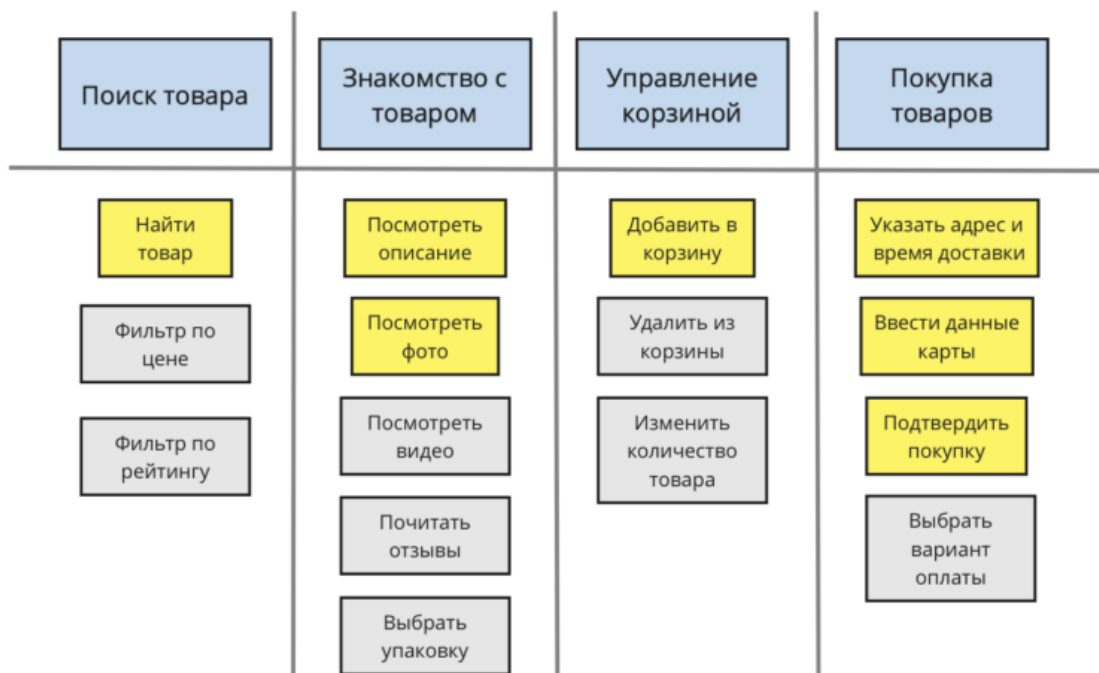
1. Расскажите историю клиента по шагам



2. Сгруппируйте действия клиента в этапы



3. Заполнение пробелов в истории



4. Приоритезируйте истории внутри каждого этапа пути

	Поиск товара	Знакомство с товаром	Управление корзиной	Покупка товаров
Сделать обязательно	Найти товар	Посмотреть описание Посмотреть фото	Добавить в корзину Удалить из корзины	Указать адрес и время доставки Ввести данные карты Подтвердить покупку
Хорошо бы сделать	Фильтр по цене	Выбрать упаковку Посмотреть видео	Изменить количество товара	Выбрать вариант оплаты
Можно будет сделать	Фильтр по рейтингу	Почитать отзывы		

5. Выделите релизы

	Поиск товара	Знакомство с товаром	Управление корзиной	Покупка товаров
Сделать обязательно	Найти товар	Посмотреть описание Посмотреть фото	Добавить в корзину Удалить из корзины	Указать адрес и время доставки Ввести данные карты Подтвердить покупку
Хорошо бы сделать	Фильтр по цене	Выбрать упаковку Посмотреть видео	Изменить количество товара	Выбрать вариант оплаты
Можно будет сделать	Фильтр по рейтингу	Почитать отзывы		

ver.1 (MVP)

ver.2

ver.3

6. Получите приоритизированный бэклог

