**Задание 1. Внимательно изучите список механик игры, которую вы создаете в рамках данного**

**курса. Каким из них вы хотите обучать игрока? Каким вы хотите учить через хард-тутор, а**

**каким - через софт-тутор?**

Для игры сбор компьютеров

**1. Механики для обучения через хард-тутор:**

* **Выбор и покупка комплектующих** — объяснить базовые характеристики деталей, совместимость и бюджетные ограничения, чтобы игрок понимал, как правильно подобрать комплектующие под заказ.
* **Пошаговая сборка компьютера** — показать последовательность установки компонентов (блок питания, материнская плата, процессор, видеокарта, память, накопители, периферия), чтобы избежать ошибок и закрепить навык.
* **Диагностика и ремонт** — научить выявлять неисправности, заменять детали и выполнять базовое обслуживание (чистка, обновление ПО).
* **Оптимизация и апгрейд** — объяснить, как настраивать охлаждение, разгонять процессоры и видеокарты, добавлять память для повышения производительности.
* **Расширение мастерской** — показать, как покупать инструменты, расширять рабочие места и нанимать помощников, чтобы ускорять процесс.

**2. Механики для обучения через софт-тутор:**

* **Продажа и выполнение заказов** — обучение через практику: игрок получает отзывы и деньги, видит последствия своих действий, что стимулирует улучшать навыки.
* **Изучение совместимости и особенностей деталей** — игрок постепенно узнаёт о нюансах в процессе игры, без навязчивых подсказок.
* **Оптимизация производительности** — игрок экспериментирует с настройками и видит результат, что способствует естественному обучению.
* **Пошаговое усложнение заказов и задач** — сложность растёт постепенно, обучая игрока на практике.

Для игры сбор программ

**1. Механики для обучения через хард-тутор:**

* Выбор и посадка фрагментов кода (функции, классы, библиотеки) — объяснение типов кода, их назначения и правильного размещения для успешного запуска проектов.
* Базовая оптимизация: отладка, рефакторинг и тестирование — пошаговое обучение, как улучшать качество и скорость разработки.
* Запуск готового проекта и сбор урожая — объяснение процесса запуска продукта, получения наград и влияния на прогресс.
* Найм и использование AI-ассистентов — обучение принципам работы помощников и их эффективному применению для ускорения разработки.

**2. Механики для обучения через софт-тутор:**

* Постепенное знакомство с разными фрагментами кода через игровой опыт, без навязчивых подсказок.
* Эксперименты с оптимизацией и тестированием, позволяющие игроку самостоятельно видеть эффект своих действий.
* Естественный процесс сбора урожая и повторения игрового цикла, мотивирующий к развитию.
* Расширение проектов за счёт объединения модулей и создания сложных систем по мере роста навыков.
* Освоение новых языков программирования и технологий через постепенное открытие и применение.
* Обучение использованию AI-ассистентов через наблюдение за их влиянием в процессе игры.

**Задание 2. Определитесь с тем, какие методы подачи обучения Вы хотите использовать в вашей**

**игре. Укажите для каждого из туториалов, каким методом вы будете подавать обучение в**

**нём.**

**Для игры «Tech Workshop Simulator» (сбор компьютеров)**

**Хард-тутор (обязательное обучение)**

| **Механика** | **Метод подачи туториала** | **Обоснование** |
| --- | --- | --- |
| Выбор и покупка комплектующих | Текст с визуалом + Контролы и действия | Объяснить характеристики и совместимость, показать выбор наглядно и дать попробовать |
| Пошаговая сборка компьютера | Контролы и действия + Текст и указатель | Пошаговое руководство с подсказками и активным взаимодействием |
| Диагностика и ремонт | Текст с визуалом + Контролы и действия | Объяснить процесс, дать возможность применить на практике |
| Оптимизация и апгрейд | Текст с визуалом + Контролы и действия | Объяснить настройку и дать попробовать изменить параметры |
| Расширение мастерской | Текст и указатель | Показать, куда нажимать и что покупать, без излишних деталей |

**Софт-тутор (обучение через игровой процесс)**

| **Механика** | **Метод подачи туториала** | **Обоснование** |
| --- | --- | --- |
| Продажа и выполнение заказов | Сами разберутся + Только визуал | Игрок учится через практику, видит результат и отзывы |
| Изучение совместимости деталей | Сами разберутся + Только визуал | Постепенное открытие нюансов без навязчивых подсказок |
| Оптимизация производительности | Сами разберутся + Только визуал | Эксперименты и наблюдение эффекта в игре |
| Пошаговое усложнение заказов | Сами разберутся + Только визуал | Рост сложности естественным образом |

**Для игры «Code Farm» (сбор программ)**

**Хард-тутор (обязательное обучение)**

| **Механика** | **Метод подачи туториала** | **Обоснование** |
| --- | --- | --- |
| Выбор и посадка фрагментов кода | Текст с визуалом + Контролы и действия | Объяснить типы кода и показать, как их размещать |
| Базовая оптимизация | Текст с визуалом + Контролы и действия | Пошагово показать отладку и тестирование |
| Запуск готового проекта и сбор урожая | Текст и указатель + Контролы и действия | Объяснить процесс запуска и награды, дать попробовать |
| Найм и использование AI-ассистентов | Текст и указатель + Контролы и действия | Показать, как нанимать и использовать помощников |

**Софт-тутор (обучение через игровой процесс)**

| **Механика** | **Метод подачи туториала** | **Обоснование** |
| --- | --- | --- |
| Постепенное знакомство с кодом | Сами разберутся + Только визуал | Игрок открывает новые фрагменты через игровой опыт |
| Эксперименты с оптимизацией | Сами разберутся + Только визуал | Позволить самостоятельно видеть эффект изменений |
| Естественный сбор урожая | Сами разберутся + Только визуал | Мотивация через игровой процесс |
| Расширение проектов | Сами разберутся + Только визуал | Постепенное усложнение и объединение модулей |
| Освоение новых языков и технологий | Сами разберутся + Только визуал | Постепенное открытие новых возможностей |
| Обучение использованию AI-ассистентов | Сами разберутся + Только визуал | Наблюдение за влиянием помощников в процессе игры |

**Задание 3. Определите: каким из видов обучающих текстов вы хотите воспользоваться. Для тех туториалов, которые, по вашему мнению, должны содержать текст.**

**Для игры «Tech Workshop Simulator»**

**Хард-тутор (обязательное обучение)**

| **Механика** | **Стиль обучающего текста** | **Обоснование** |
| --- | --- | --- |
| Выбор и покупка комплектующих | Интерфейсные | Чёткие инструкции и описания характеристик в UI |
| Пошаговая сборка компьютера | Диегетические + Интерфейсные | Инструкции от персонажа или визуальные подсказки в мире + UI |
| Диагностика и ремонт | Диегетические | Подсказки от клиента или помощников, встроенные в сюжет |
| Оптимизация и апгрейд | Интерфейсные | Технические подсказки в UI, объясняющие параметры |
| Расширение мастерской | Диегетические | Встроенные в мир подсказки (например, советы коллег) |

**Софт-тутор (обучение через игровой процесс)**

| **Механика** | **Стиль обучающего текста** | **Обоснование** |
| --- | --- | --- |
| Продажа и выполнение заказов | Сами разберутся | Игрок учится через практику, без текстовых подсказок |
| Изучение совместимости деталей | Сами разберутся | Постепенное открытие знаний без прямого текста |
| Оптимизация производительности | Сами разберутся | Эксперименты и наблюдение результатов |
| Пошаговое усложнение заказов | Сами разберутся | Рост сложности естественным образом |

**Для игры «Code Farm»**

**Хард-тутор (обязательное обучение)**

| **Механика** | **Стиль обучающего текста** | **Обоснование** |
| --- | --- | --- |
| Выбор и посадка фрагментов кода | Интерфейсные | Чёткие описания и инструкции в UI |
| Базовая оптимизация | Интерфейсные | Пошаговые подсказки в интерфейсе |
| Запуск готового проекта и сбор урожая | Диегетические + Интерфейсные | Встроенные в мир объяснения (например, AI-ассистент говорит) + UI подсказки |
| Найм и использование AI-ассистентов | Диегетические | Персонажи или AI-ассистенты дают советы в игровом мире |

**Софт-тутор (обучение через игровой процесс)**

| **Механика** | **Стиль обучающего текста** | **Обоснование** |
| --- | --- | --- |
| Постепенное знакомство с кодом | Сами разберутся | Игрок открывает новые возможности самостоятельно |
| Эксперименты с оптимизацией | Сами разберутся | Свободное изучение без текстовых подсказок |
| Естественный сбор урожая | Сами разберутся | Мотивация через игровой процесс |
| Расширение проектов | Сами разберутся | Постепенное усложнение без прямого текста |
| Освоение новых языков и технологий | Сами разберутся | Постепенное открытие новых возможностей |
| Обучение использованию AI-ассистентов | Сами разберутся | Наблюдение за эффектом помощников |

**Задание 4. Пишем первый текст! Выберите один из туториалов, который должен содержать текст. Напишите текст в выбранном вами стиле**

Для игры сбор компьютера

Стиль текста — **диегетический** (встроенный в игровой мир, от лица наставника-мастера), чтобы сохранить погружение и ощущение, что игрок получает советы от друга и коллеги.

**Текст туториала: Пошаговая сборка компьютера (диегетический стиль)**

Привет! Добро пожаловать в мастерскую. Сегодня мы вместе соберём первый компьютер. Не переживай, я помогу на каждом шаге.

1. Сначала устанавливаем блок питания — он даст энергию всему ПК. Просто поставь его в корпус и закрепи винтами.
2. Теперь материнская плата — сердце системы. Аккуратно вставь её в корпус и подключи к блоку питания.
3. Следующий шаг — процессор. Осторожно открой сокет, вставь процессор и защёлкни фиксатор.
4. Не забудь про оперативную память — вставь планки в слоты на материнской плате до щелчка.
5. Видеокарта — для красивой картинки и игр. Вставь её в соответствующий слот и закрепи.
6. Подключаем накопители — жёсткий диск или SSD — туда, где есть специальные крепления.
7. Проверь все кабели: питание, данные и вентиляторы. Всё должно быть надёжно и аккуратно.

Если что-то непонятно — не спеши, повтори шаг или спроси. Главное — делать всё внимательно и с удовольствием. Вперед, твой первый ПК уже почти готов!

Для игры посадка кода

Добро пожаловать в Code Farm!  
Здесь вы «сажаете» фрагменты кода — функции, классы и библиотеки. Каждый из них — кирпичик будущего приложения.

* Выберите фрагмент кода из списка.
* Нажмите «Посадить», чтобы начать его развитие.
* Следите за ростом — со временем код превратится в полноценный модуль.

Не забывайте: правильный выбор фрагментов ускорит создание проектов и принесёт больше прибыли. Попробуйте сейчас!

**Задание 5. Опишите контекст для написанного вами туториального текста**

**Игра «Tech Workshop Simulator» — Пошаговая сборка компьютера**

**1. Зачем игроку это нужно?**  
Чтобы собрать рабочий компьютер и выполнить первый заказ, получить деньги и репутацию.

**2. Почему ему должно быть интересно?**  
Игрок становится настоящим мастером, который создаёт уникальные ПК, решает задачи клиентов и развивает мастерскую.

**3. Как он дошёл до жизни такой?**  
Начинающий мастер, только что открыл свою мастерскую и получил первый заказ на сборку.

**4. Как краткосрочная цель скажется на долгосрочной?**  
Успешная сборка первого компьютера откроет доступ к новым заказам и улучшениям мастерской.

**5. Почему именно игрок должен это делать?**  
Потому что он — мастер, который отвечает за качество и скорость работы.

**6. Что будет, если игрок этого не сделает?**  
Заказ не будет выполнен, мастерская не заработает деньги и не сможет расти.

**7. Почему игрок может это сделать?**  
Потому что у него есть инструменты, инструкции и поддержка наставника.

**Контекст через монолог наставника (диегетический стиль)**

«Добро пожаловать в мастерскую! Сегодня мы вместе соберём твой первый компьютер. Это важный шаг — от него зависит успех твоей мастерской. Я помогу тебе разобраться, как правильно устанавливать каждую деталь, чтобы всё работало без сбоев. Если сделать всё внимательно, клиенты останутся довольны, и ты сможешь развиваться дальше. Не бойся — вместе справимся!»

**Игра «Code Farm» — Выбор и посадка фрагментов кода**

**1. Зачем игроку это нужно?**  
Чтобы начать создавать программные проекты и зарабатывать ресурсы для развития студии.

**2. Почему ему должно быть интересно?**  
Игрок становится создателем уникальных приложений, управляет процессом разработки и строит IT-империю.

**3. Как он дошёл до жизни такой?**  
Начинающий разработчик с маленькой студией, который хочет вырасти в крупную компанию.

**4. Как краткосрочная цель скажется на долгосрочной?**  
Правильный выбор и посадка кода ускорит рост проектов и откроет новые возможности.

**5. Почему именно игрок должен это делать?**  
Потому что он — руководитель и главный разработчик, принимающий ключевые решения.

**6. Что будет, если игрок этого не сделает?**  
Проекты будут расти медленно или неэффективно, что замедлит развитие студии.

**7. Почему игрок может это сделать?**  
Потому что у него есть доступ к фрагментам кода и инструменты для их развития.

**Контекст через монолог рассказчика (интерфейсный стиль)**

«Добро пожаловать в Code Farm! Здесь вы начинаете с простых фрагментов кода — функций, классов и библиотек. Правильно выбирайте, что посадить первым: от этого зависит скорость роста ваших проектов. С каждым успешным запуском вы будете продвигаться к созданию сложных и прибыльных приложений. Управляйте умело, и ваша студия станет лидером в индустрии!»