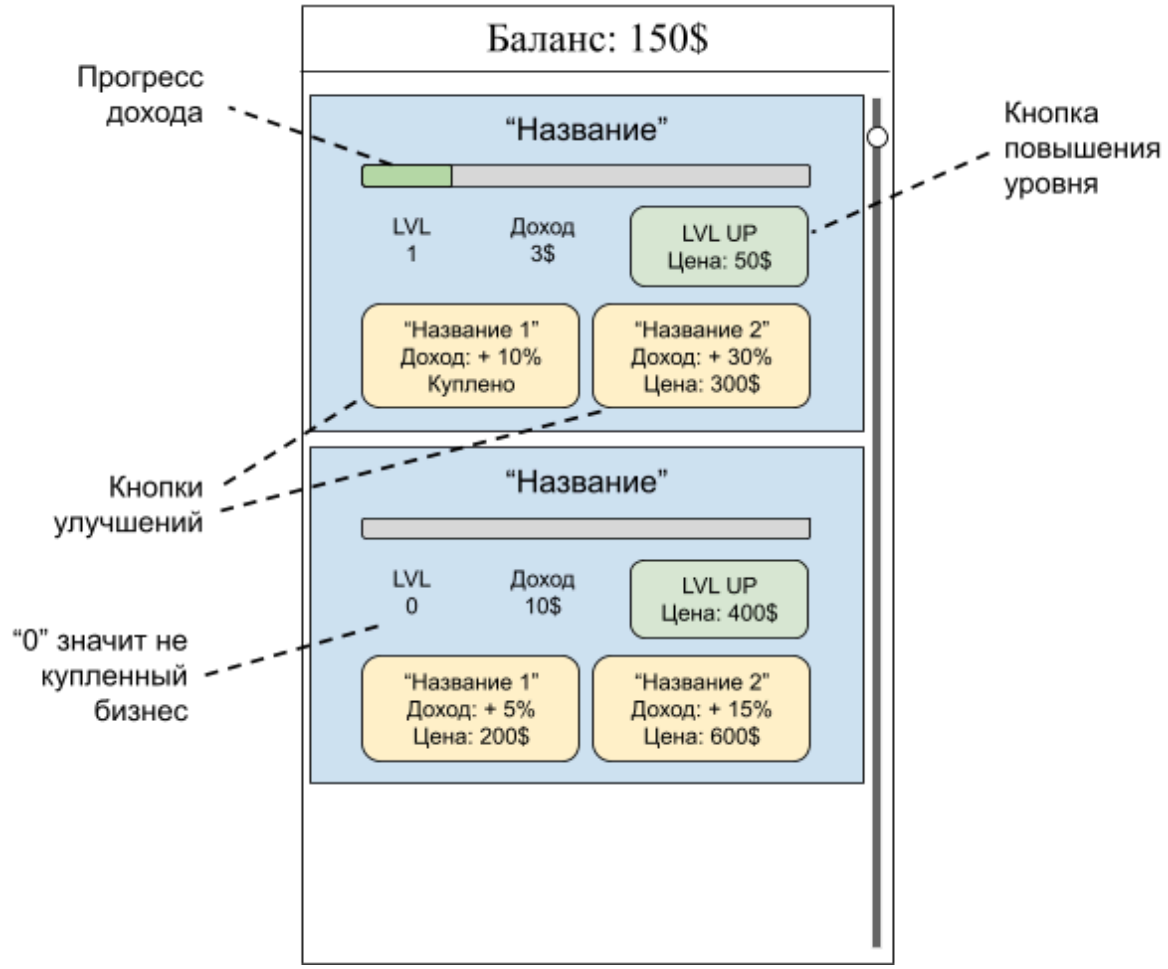
Платформа Android. Unity 2022+



Задача

Кандидату предлагается сделать упрощенный прототип кликера, в котором игрок покупает и улучшает бизнесы, генерирующие доход.

Баланс игрока

Отображается вверху экрана и не может опуститься ниже 0$.

Бизнесы

Каждый бизнес изображается отдельным элементом в списке на экране и имеет:

1. Название
2. Текущий уровень

   Если бизнес не куплен, то там отображается 0.
3. Текущий доход

4. Прогресс дохода

   Бар заполняется от 0% до 100% за время “Задержка дохода” из конфига ниже. Как только доходит до 100%, значение текущего дохода зачисляется в Баланс, а прогресс начинает копиться заново с 0%. Значение бара должно обновляться каждый кадр.
5. Кнопка повышения уровня

   Содержит цену повышения уровня.

   При нажатии покупается уровень.
6. Кнопка улучшения 1

   При нажатии улучшение покупается.
7. Кнопка улучшения 2

   При нажатии улучшение покупается.

Улучшения бизнесов

Каждое улучшение уникально для каждого бизнеса. Каждое можно купить только 1 раз. Каждое обладает:

1. Название
2. Цена покупки

   После покупки, заменяется на текст “Куплено”.
3. Множитель дохода (в процентах)

Конфиг

Доход = lvl * базовый_доход * (1 + множитель_от_улучшения_1 + множитель_от_улучшения_2)

Цена уровня = cost = (lvl+1) * базовая_стоимость

Бизнес	Задержка дохода	Базовая стоимость	Базовый доход	Улучшение 1		Улучшение 2	
				Цена	Множитель дохода	Цена	Множитель дохода
Бизнес 1	3 сек	3	3	50	+50%	400	+100%
Бизнес 2	6 сек	40	40	1200	+100%	4000	+200%
Бизнес 3	10 сек	200	200	6000	+100%	20000	+150%
Бизнес 4	17 сек	1000	1000	15000	+100%	50000	+200%
Бизнес 5	30 сек	5000	5000	100000	+200%	500000	+400%

Важные моменты

1. В списке доступно 5 бизнесов. Первый бизнес изначально куплен, остальные нет. Чтобы купить бизнес, игрок должен купить его первый уровень.
2. Все текстовые поля изменяются в момент изменения значений и должны содержать актуальную информацию (например, цена покупки уровня изменяется после покупки и тд).
3. Игра обязательно должна сохраняться при выходе из игры. Прогресс дохода у каждого бизнеса тоже.
4. Конфиг (или конфиги) должен быть реализован через ScriptableObject.
5. Названия бизнесов и улучшений должны подтягиваться из другого ScriptableObject.

Требования к коду

1. Использовать только LeoEcs/LeoEcsLite, без расширений.
2. Не использовать DI фреймворки (Zenject, VContainer и тд), зависимости протаскивать через конструкторы или Init методы.
3. Не использовать какие-либо сторонние/самописные фреймворки, помимо обозначенных ECS библиотек.
4. Придерживаться код-стайла Microsoft.

Результат и оценка

В первую очередь оценивается ECS архитектура, соблюдение код-стайла, отсутствие костылей (даже если они быстро исправляются), работоспособность прототипа, внимательность к заданию.

Всё указанное в “важных моментах” и “требованиях к коду” должно быть соблюдено.

В дополнительных ассетах (спрайты, звуки) нет нужды, они никак не повлияют на оценку.

Расширяемость кода обеспечивать не нужно, выстраивать архитектуру сверх необходимого не нужно, лишний код негативно повлияет на оценку.

Кандидату необходимо залить исходники на github и предоставить ссылку с публичным доступом. Так же, необходимо собрать apk файл и предоставить ссылку на него отдельно. Указать затраченное время.