

Привет! В прошлый раз забыл обратить твое внимание на этот файл, поэтому пояснения по прошлой неделе не очень актуальны, но будет здорово если найдется пара минут взглянуть:

13-я неделя.

Задержку с показом карт реализовал добавив в класс `EffectsManager` метод `ShowCardsWithDelay`, который вызывает метод `ShowCards` через `Invoke` с задержкой в 1.2 секунды. Такое вот решение “в лоб”.

Для запуска эффекта повышения уровня в класс `Collector` (единственный, который знает о положении игрока, т.к. является его компонентом) добавил поле `_levelUpParticlesPrefab`, куда в инспекторе я назначил префаб этого эффекта. Префаб инстанцируется в методе `OnLevelUp()`, являющийся коллбэком (если я правильно называю) события `LevelUpEvent`, запускающимся в классе `ExperienceManager` при повышении уровня игрока.

Круг вокруг игрока сделал при помощи `Handles` класса `OnDrawGizmos`, нарисовал в двух плоскостях для большей наглядности.

В процессе возникло пару проблем: при повышении уровня и показе карт игрок мог продолжать движение, оставляя за собой частицы эффекта повышения уровня. Для его остановки в класс `RigidbodyMove` я добавил поле `_isMovable`, значение которого задается в публичном методе `SetMoveability` и проверяется в `Update` при обработке ввода. На `false` значение этого поля меняется в методе `OnLevelUp` класса `Collector`. Понимаю, что так себе решение, уверен что на следующей неделе на уроках это будет реализовано по-другому.

С возвращением полю значения `true` при клике на карту сложность была с тем, что карта и игрок не связаны напрямую. Класс `Collector` связан с `ExperienceManager`, который связан с `EffectManager`, ссылка на который, в свою очередь, есть у каждого экземпляра класса `Card`. Мне было интересно сделать все не создавая новые поля в классах и назначая ссылки в инспекторе, поэтому цепочка получилась длинная и сбивающая с толку, но тоже вроде бы рабочая: при клике на карте вызывается метод `Click`, в котором в `EffectManager` запускается событие `EffectHasBeenAdded`. На это событие подписан `ExperienceManger` методом `OnEffectAdded`, в котором, в свою очередь запускается событие `EffectAddedEvent`, на которое подписан `Collector`.

14-я неделя.

Добавил эффект `BlackHole`, при входе в коллайдер которого у врага отключается компонент, отвечающий за его перемещение и отключаются ограничения `Rigidbody`.

Было интересно разобраться с работой корутин, поэтому я добавил пару для косметических эффектов:

- При смерти игрока его коллайдер увеличивается в размере, враги как бы расступаются вокруг во время проигрывания анимации смерти.
- Жизненный цикл префаба черной дыры сделал через корутину, запускающуюся в `Start`'е скрипта `BlackHole.cs`.
- Заполнение `bar`'ов здоровья и опыта тоже сделал через корутины для плавности.

Не получилось с частицами при смерти врага (скрипт `Enemy.cs`, строка 97). Метод срабатывает, ссылка на частицы в префабе есть, но почему-то не работает. Если инстанцировать частицы в включенной галочкой “`Play on Awake`”, то работает, но в прошлой проверке ты говорил, что не сторонник такого подхода. Помоги, пожалуйста, разобраться.

Надеюсь, пояснения не запутали еще больше)). Заранее спасибо за проверку ДЗ.