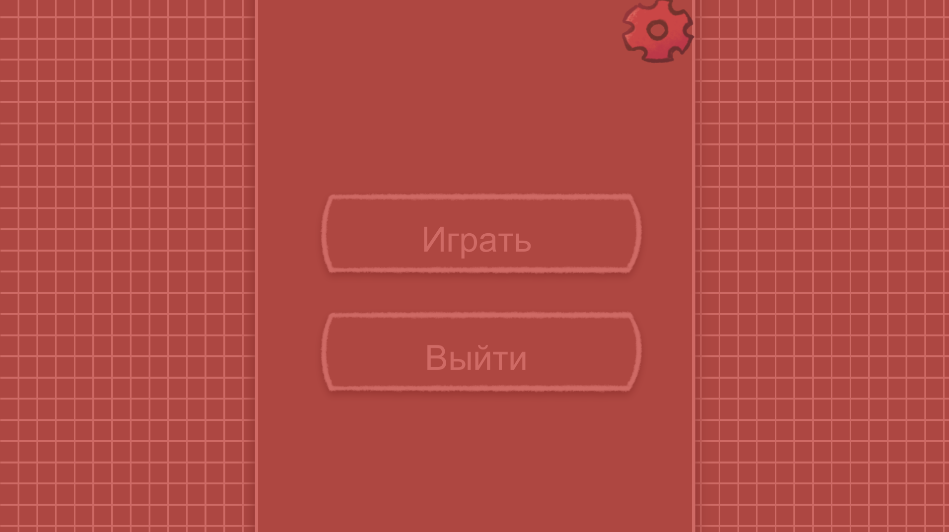
## Подготовка спрайтов и текстур для будущего приложения

Для этой программы все спрайты были отрисованы вручную (см. Рис. 1.). Нередко начинающие разработчики прибегают к помощи готовых пакетов спрайтов и текстур, что может ускорить процесс обучения работы с новой платформой для создания игр. Однако, часто бывает такое, что среди готовых пакетов нельзя найти подходящий, в таком случае обычно разработчики прибегают к помощи дизайнеров или отрисовывают что-то сами.

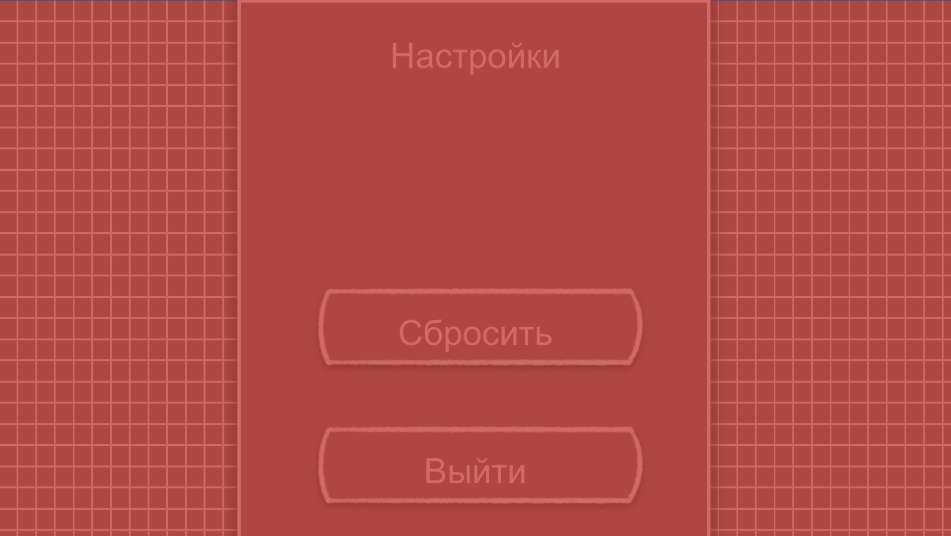
**Рис.1. «Готовые спрайты и текстуры для приложения»**

## Интерфейс приложения и игровой процесс

Для дизайна начального экрана входа был выбран простой и понятный дизайн с двумя основными кнопками: “Играть” и “Выйти” (см. Рис. 2.). 

**Рис.2. «Интерфейс начального экрана входа»**

В верхнем правом углу располагается иконка кнопки настроек. Здесь (см. Рис. 3.) также есть всего две кнопки: “Выйти”, имеется в виду из настроек, и “Сбросить”. Последняя кнопка предназначена для сброса игровых параметров в случае проигрыша или при иной необходимости.



**Рис.3. «Интерфейс панели настроек»**

По нажатию на кнопку “Играть” приложение переходит на основную игровую сцену, комнату питомца (см. Рис. 4.), где и находится основной интерфейс для ухода за питомцем и сам питомец.



**Рис.4. «Интерфейс комнаты питомца»**

Как уже было сказано, в центре экрана и находиться питомец. В зависимости от текущих показателей, а именно: “Сытость”, “Счастье”, “Энергия”, внешний вид питомца будет меняться.

Например, если уровень “Счастья” опуститься ниже 50, он будет выглядеть немного грустным (см. Рис. 5.). Если уровень опуститься ниже 20, питомец станет очень грустным (см. Рис. 6.). Если пользователь видит одно из этих состояний, то это должно дать ему сигнал к тому, чтобы поиграть с питомцем. Для этого нужно нажать на кнопку посредине нижней панели со всеми кнопками.



**Рис.5. «Состояние питомца “Грустный”»**



**Рис.6. «Состояние питомца “Очень грустный”»**

В случае если показатель “Энергии” упадет ниже 20, питомец будет выглядеть уставшим (см. Рис. 6.), что должно дать сигнал игроку, уложить питомца спать. Для этого нужно нажать на предпоследнюю кнопку на нижней панели со всеми кнопками.



**Рис.6. «Состояние питомца “Сильная усталость”»**

В состоянии сна (см. Рис. 7.) показатель “Энергии” будет постепенно увеличиваться. В этом состоянии питомца нельзя кормить или играть с ним.



**Рис.7. «Состояние питомца “Спит”»**

Если уровень “Сытости” опуститься ниже 50, питомец будет выглядеть немного голодным (см. Рис. 8.). Если уровень опуститься ниже 20, питомец станет очень голодным (см. Рис. 9.). Если пользователь видит одно из этих состояний, то это должно дать ему сигнал к тому, чтобы покормить питомца. Для этого нужно нажать вторую кнопку на нижней панели со всеми кнопками.



**Рис.8. «Состояние питомца “Голодный”»**



**Рис.9. «Состояние питомца “Очень голодный”»**

При кормлении питомца расходуется единица еды, пополняющая показатель сытости на 30 процентов от максимального показателя. Кол-во оставшейся еды можно посмотреть в окне инвентаря справа от питомца.

Когда еда закончится ее нужно будет купить в магазине, который можно найти слева от питомца. При покупке расходуются монеты. Когда они закончатся пользователь столкнется с проблемой: показатель сытости продолжает опускаться, а еда постепенно заканчивается, и, если не найти еще монет, показатель сытости опуститься полностью, что приблизит питомца к “гибели”, а игрока, следовательно, к проигрышу.

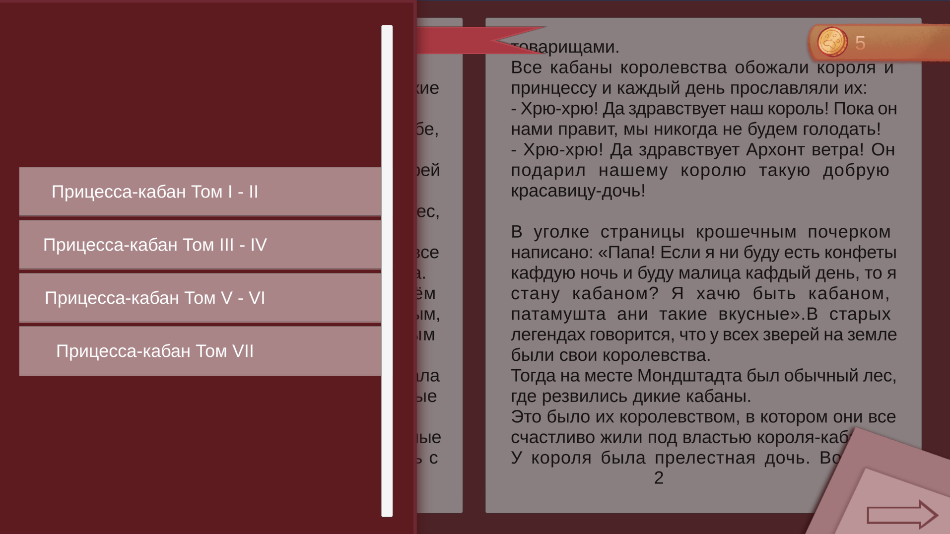
Здесь игроку на помощь приходит встроенная читалка, в которую можно перейти по нажатию на первую кнопку на нижней панели, и в которой за чтение имеющихся в “Библиотеке” книг, то есть продвижение по книге вперед, можно получать еще монеты, а именно 5 монет за страницу.

Читалка (см. Рис. 10.) выполнена в стиле классической книги с двумя страницами, одна слева, другая справа. Чтобы перемещаться дальше по книге, достаточно нажать на правую страницу, а чтобы вернуться на предыдущую, нужно нажать на левую страницу.



**Рис.10. «Экран читалки с открытой первой в “Библиотеке” имеющейся книгой»**

Чтобы перейти в “Библиотеку” с остальными книгами, нужно нажать на красную ленту в верхнем левом углу.



**Рис.11. «Экран “Библиотеки”»**

Здесь можно выбрать другую книгу из имеющихся. По нажатию на одну из панелей с названием книги, содержание на форме с книгой поменяется на содержание выбранной книги. Чтобы закрыть “Библиотеку” (см. Рис. 11.), достаточно нажать снова на красную ленту, которая теперь располагается наверху сбоку панели со списком книг. Также, в верхнем правом углу в окне “Библиотеки”, можно посмотреть текущее число монет.

Чтобы покинуть читалку, нужно в окне “Библиотеки” нажать на кнопку в нижнем правом углу. Игра возвращает пользователя в комнату питомца, где его можно будет уже покормить, купив на накопленные монеты еды.

Однако, если игрок пренебрежет не только “Сытостью” питомца, но еще “Энергией” и “Счастьем”, вскоре все показатели упадут до нуля и питомец “погибнет” (см. Рис. 12.), что приведет к потере доступа ко всем кнопкам без возможности мгновенно возродить любимца.

Чтобы начать заново, нужно, по нажатию на последнюю кнопку на нижней панели, выйти обратно на начальный экран. Затем зайти в настройки и нажать “Сбросить”.

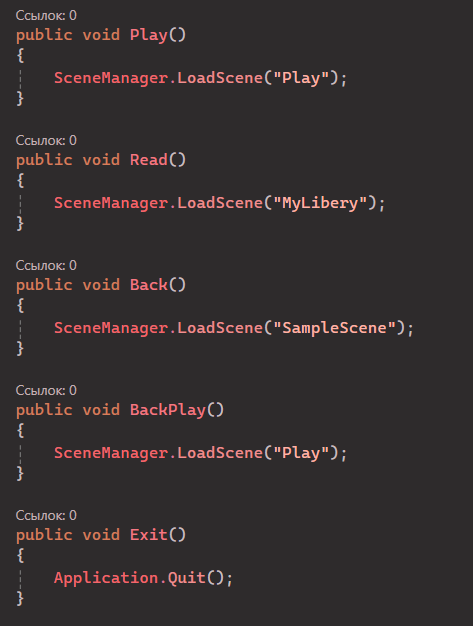


**Рис.12. «Проигрыш»**

По возвращению в комнату питомца, он вернется в начальное состояние, все монеты будут потеряны и купленная еда тоже.

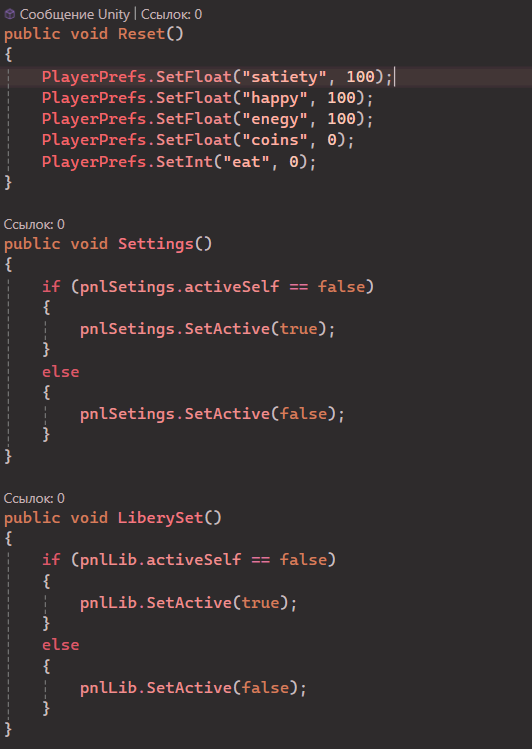
# Реализация программного кода приложения игры в стиле “Тамагочи”

Реализацию кода, после подготовки интерфейса, следует начать с перехода по сценам. По нажатию по одной из кнопок: “Играть”, кнопка для перехода в читалку, кнопка для возвращения на начальный экран, кнопка для возвращения в игру из “Библиотеки”, пользователь будет перемещаться по сценам. А последняя процедура предназначена для кнопки выхода из игры (см. Рис. 13.).



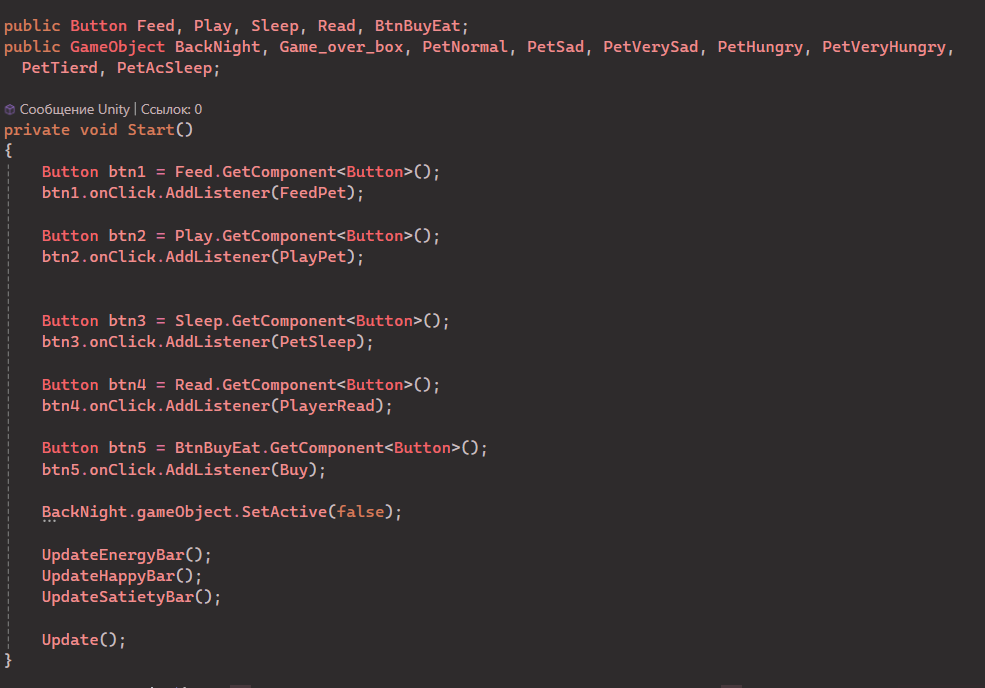
**Рис.13. «Реализация кода для перехода между сценами и выхода из игры»**

Далее для кнопки сбросить параметры пишется функция, изменяющая переменные для PlayerPrefs, который позволяет сохранять данные о показателях, а именно: числе монет и еде, после выхода из приложения. Все показатели питомца возвращаются к сотне, число монет обнуляется и число единиц еды тоже. Далее код для показа или скрытия панели с настройками, на начальном экране, и панели “Библиотеки” в читалке (см. Рис. 14.).

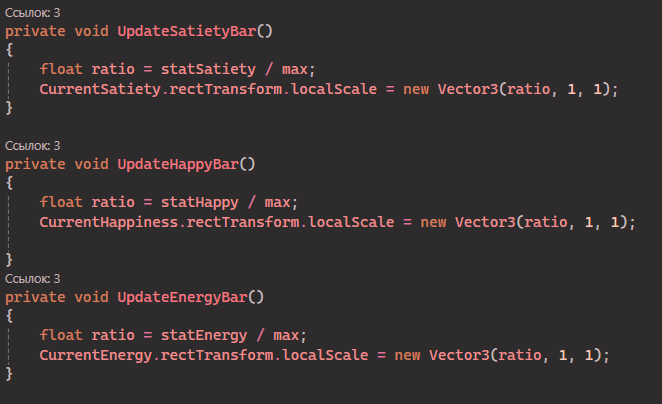


**Рис.14. «Реализация кода для кнопок “Сбросить”, показа настроек и показа “Библиотеки”»**

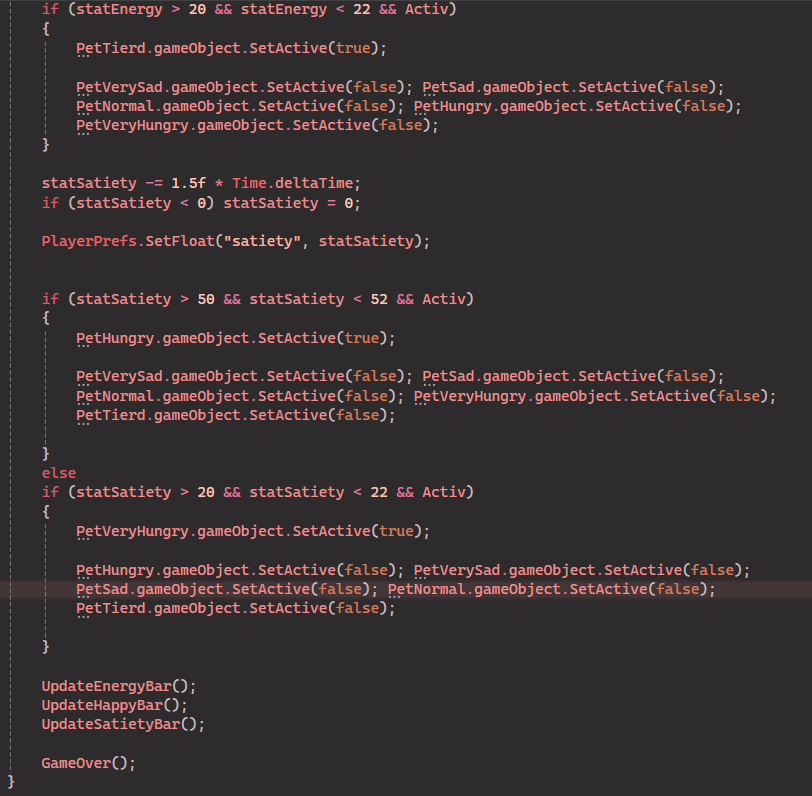
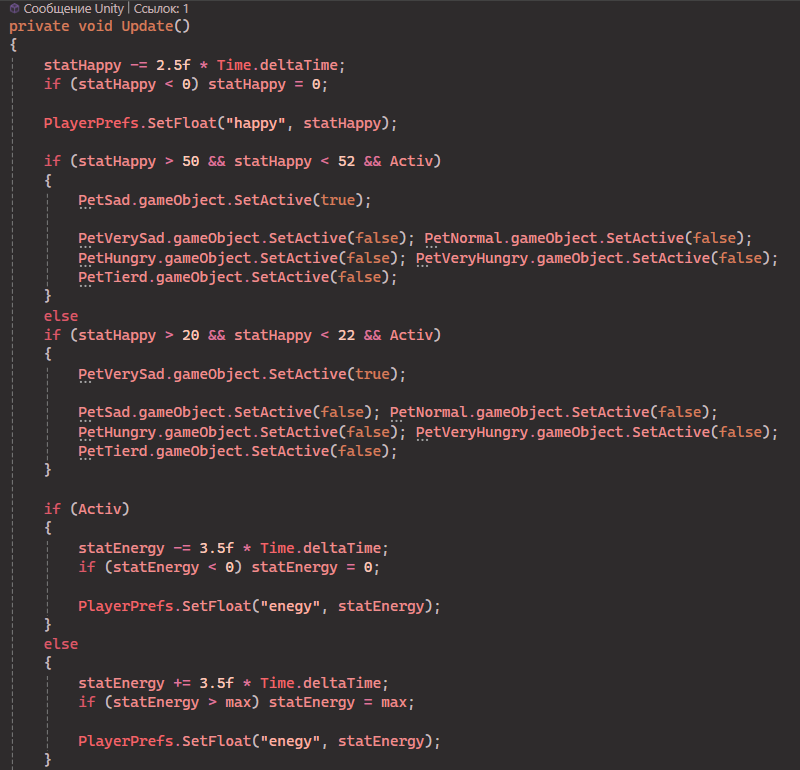
Далее при загрузке сцены с питомцем (см. Рис. 15.), в коде именуема сцена “Play”, кнопкам присваиваются соответствующие события. А также сразу обновляются состояния статус-баров (см. Рис. 16.) для показателей питомца. Последняя, Update, является стандартной для приложений Unity, срабатывающая раз за кадр, что позволяет постоянно обновлять показатели во время нахождения пользователя на этой сцене.



**Рис.15. «Реализация кода для кнопок на начальном экране и “Библиотеке”»**

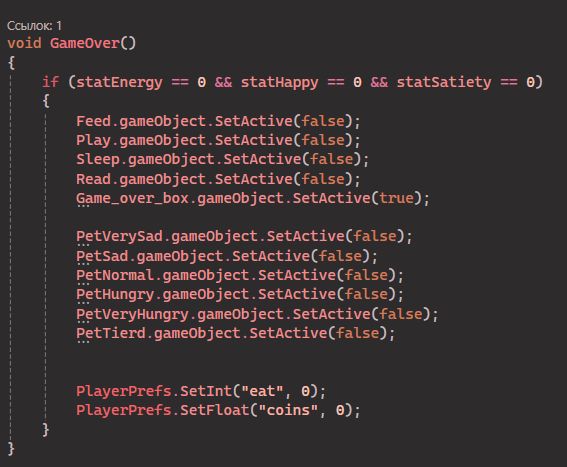
****

**Рис.16. «Реализация кода для обновления длинный статус-баров»**

В функции Update (см. Рис. 17.) с определенной периодичностью все показатели уменьшаются на небольшое число процентов, при этом сразу после уменьшения происходит проверка: если показатель проходит указанную границу внешний вид питомца меняется в соответствии с пересеченной границей. Например если в определенный момент показатель “Счастья” проходит границу “меньше 50”, то внешний вид меняется на “Грустный”. 

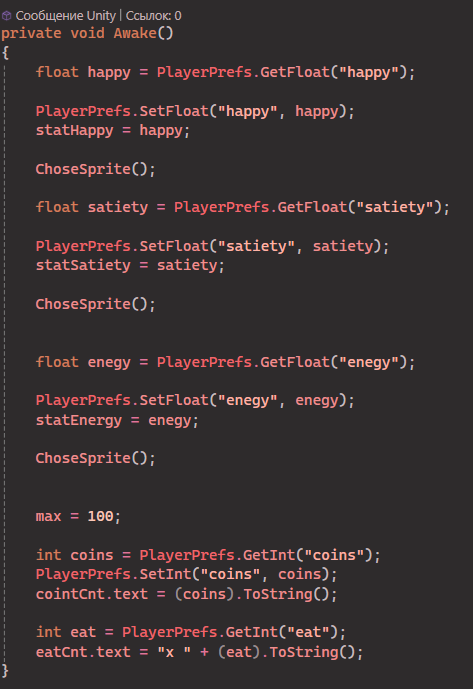
**Рис.17. «Функция Update »**

В конце функции обновляем статус-бары и запускаем функцию Game over (см. Рис. 18.) для проверки, не изменились ли все показатели питомца до 0, если да, принять соответствующие действия.

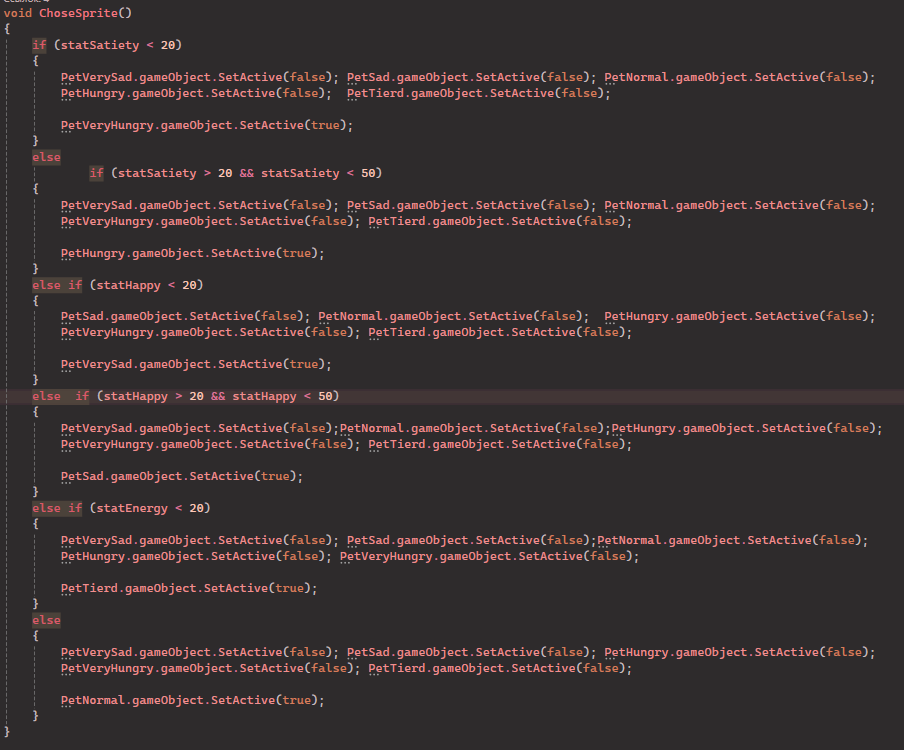


**Рис.18. «Функция GameOver»**

Чтобы при выходе из программы и повторном входе показатели питомца сохранились, в Update есть сохранение данных в переменные PlayerPref, а в стандартной функции Awake (см. Рис. 19.) происходит чтение и установка этих данных, а также установка соответствующего внешнего вида для питомца(см. Рис. 20.).



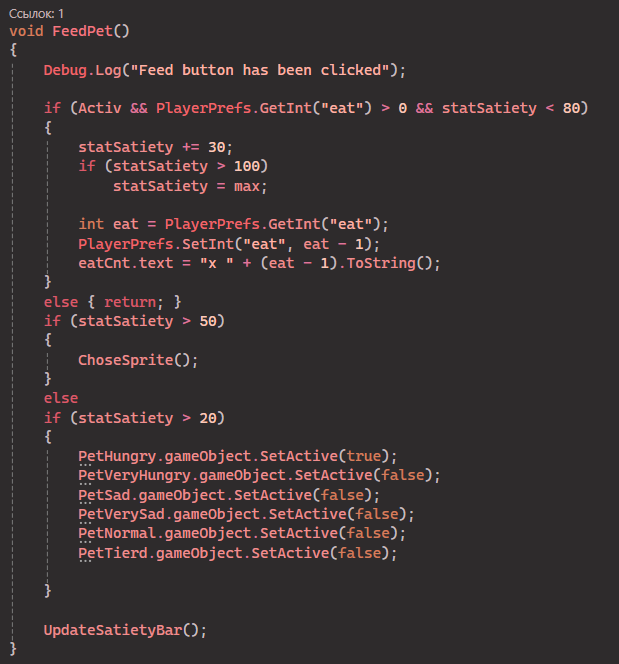
**Рис.19. «Функция GameOver»**



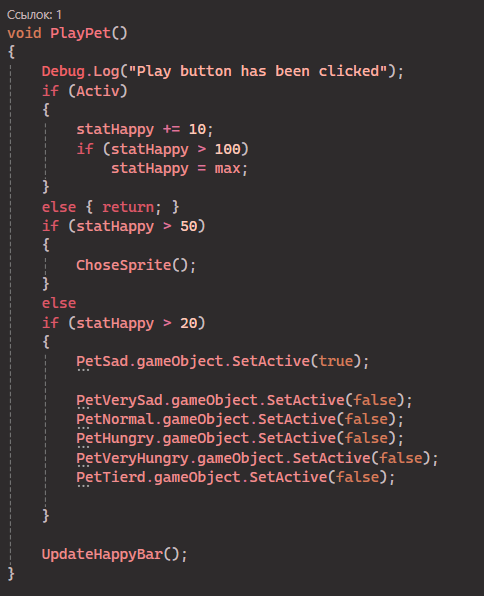
**Рис.20. «Функция ChoseSprite»**

Далее написание функций для действий которые будут происходить по нажатию на кнопки.

Кнопка покормить (см. Рис. 21.) проверяет есть ли в инвентаре еще мясо, если да, показатель увеличивается, а кол-во еды уменьшается на один. Далее, в случае, если показатель сытости больше 50, с помощью ChoseSprite приложение сверяется с остальными показателями и, случае каких-то отклонений или подтверждении нормы, устанавливает соответствующий внешний вид питомцу. Аналогично для кнопки для игры с питомцем (см. Рис. 22.).

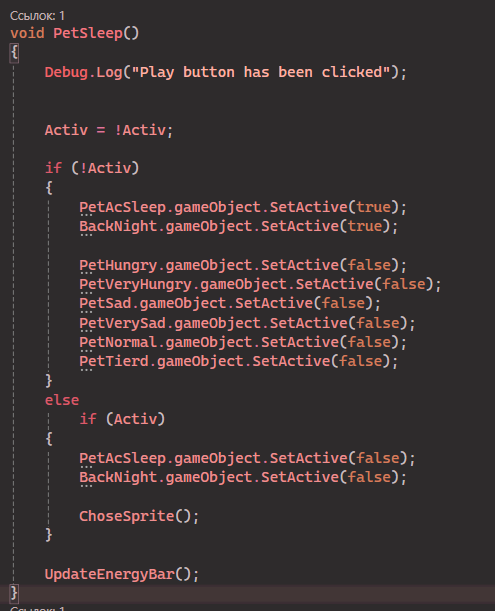


**Рис.21. «Функция FeedPet»**



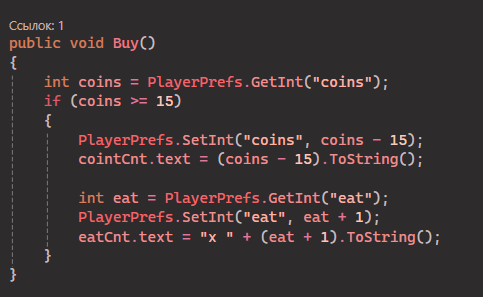
**Рис.22. «Функция PlayPet»**

На функции сна (см. Рис. 23.), помимо внешнего вида питомца, меняется и фон. Также отключается доступ ко всем кнопкам кроме чтения и выхода в меню.



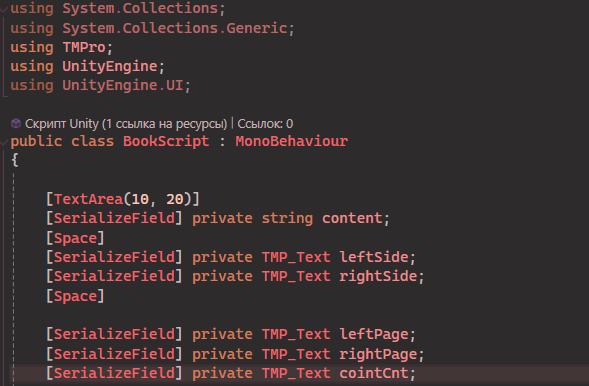
**Рис.23. «Функция PetSleep»**

Для покупки еды используется функция Buy (см. Рис. 24.), в которой проверяется, есть ли у пользователя достаточно монет, и если есть, отнимает определенное количество, добавив в инвентарь единицу еды и сохранив параметры баланса и еды.



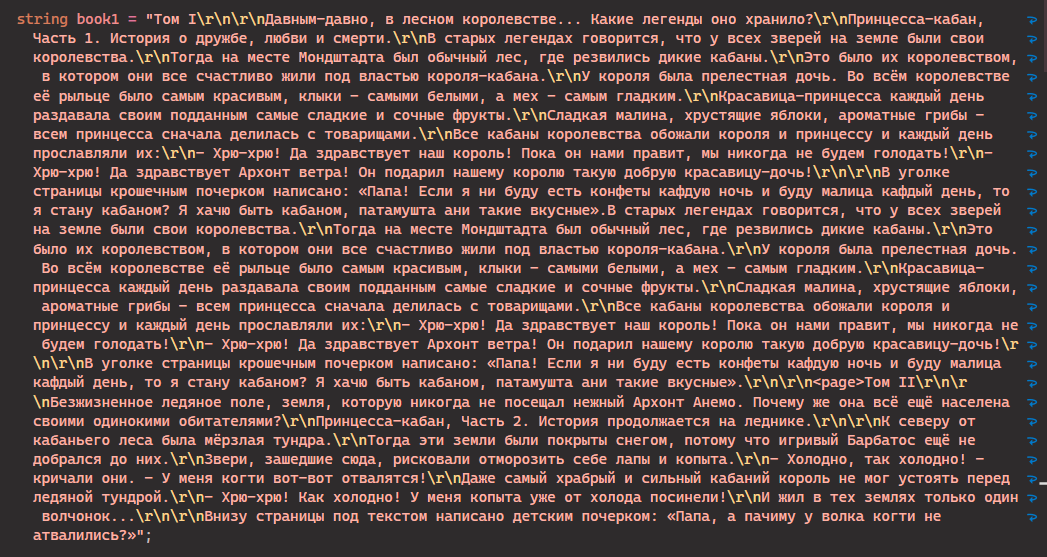
**Рис.24. «Функция Buy»**

Далее код для работы с читалкой(см. Рис. 25.). Так как она была реализована с использованием такого объекта как TextMeshPro, то для начала необходимо перед написанием кода подключить библиотеку using TMPro и, предварительно изучив документацию по работе с этим объектом, объявить необходимые переменные.



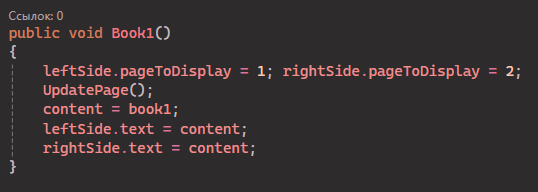
**Рис.25. «Подключение библиотеки using TMPro и объявление переменных»**

Далее загрузка содержимого книги в строковую переменную (см. Рис. 26.). Лучше копировать текст сразу в кавычки, так будут сохранены все переносы и разрывы страниц.



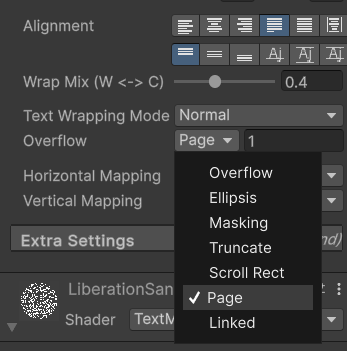
**Рис.26. «Помещенное в текстовое поле содержание книги»**

Далее помещается содержимое переменной в переменную content (см. Рис. 27.), в которой и должен храниться текст, который в будущем будет дублироваться на каждую сторону книги, а, в зависимости от указанной страницы, на стороне будет указываться соответствующая часть.



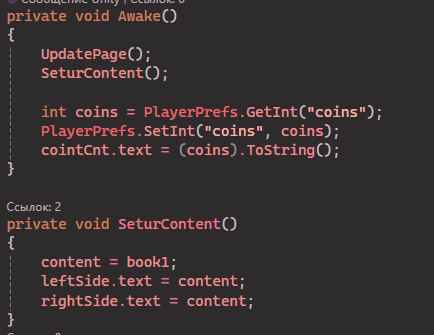
**Рис.27. «Добавление содержимого в поле с контентом для страниц и установка соответствующих номеров страниц»**

Следует отметить, что для того, чтобы текст в текстовом поле типа TextMechPro отображался постранично, нужно выбрать соответствующий режим в свойсвах объекта (см. Рис. 28.).



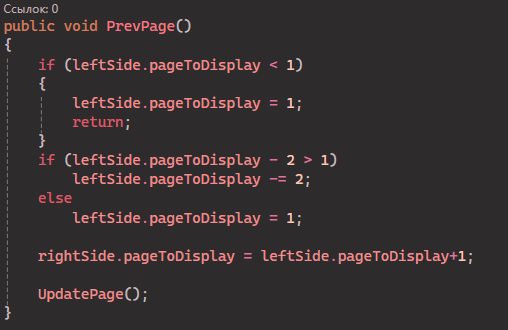
**Рис.28. «Изменение свойств отображения содержимого текстового поля со страницей на тип Page»**

При загрузке сцены в переменную с числом монет сразу заноситься текущее значение, а в переменную контент помещается содержание первой имеющейся книги (см. Рис. 29.).



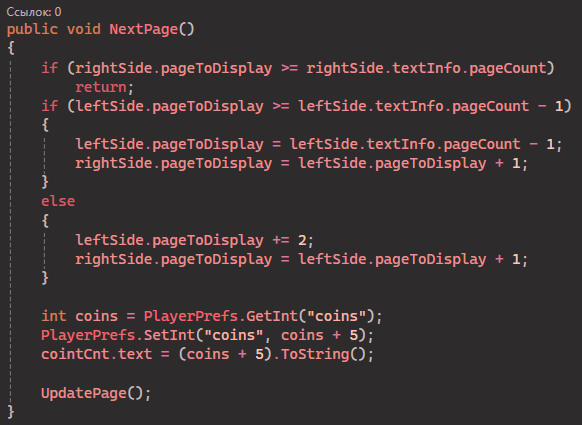
**Рис.29. «Загрузка первой книги»**

При перелистывании страниц (см. Рис. 30.) содержание текстового поля со страницей не меняется, меняется лишь отображаемая страница. Доступ к перелистыванию вперед (см. Рис. 31.) есть, пока не будет встречена последняя страница.

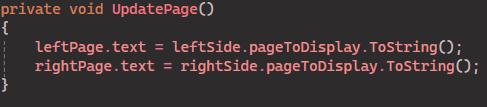


**Рис.30. «Функция возвращения на предыдущую страницу»**

Также внизу в страницы есть текстовое поле с номером который тоже изменяется в соответствии с изменениями (см. Рис. 32.).



**Рис.31. «Функция перемещения на следующую»**



**Рис.32. «Функция изменения номера страницы»**

При перелистывании на следующую страницу за одно перелистывание число монет увеличивается на 5 и сразу сохраняется.