

Домашняя работа №1



Кошка для программиста

Описание задачи

Version <2.0>

Содержание

ТРЕБОВАНИЯ К ПРИЛОЖЕНИЮ	3
ОБЩЕЕ ОПИСАНИЕ ПРОЦЕССА.....	3
ФУНКЦИОНАЛЬНЫЕ ТРЕБОВАНИЯ.....	4
Требования к интерфейсу пользователя	4
СРЕДА ВЫПОЛНЕНИЯ, ИНСТРУМЕНТЫ, ТЕХНОЛОГИИ	4
Клиент часть	4
ТРЕБОВАНИЯ К РЕАЛИЗАЦИИ	4
Основные	4
Дополнительные.....	4

Требования к приложению

Создать консольное приложение для зоомагазинов, которое обеспечит вывод информации о приобретенном электронном домашнем питомце, а также производить манипуляции с ним.

Общее описание процесса

Основные сущности: Кошка.

Клиент зоомагазина может приобрести консольное приложение, которое позволит ему имитировать наличие домашней кошки. Приложение позволит дать имя кошке, при этом имя задается хозяином в любое время после приобретения, но изменить его нельзя. Хозяин сможет кормить и наказывать своего питомца. Узнавать самочувствие кошки, по состоянию ее окраса. В момент покупки, электронное животное должно «активироваться» с заданным возрастом. Будущий хозяин обязан определить какого возраста должна быть приобретаемая кошка и в дальнейшем он не имеет права сделать ее моложе или старше.

Взаимосвязи:

- В программе один доменный объект - Cat, который подключается в консольный проект и используется в классе Program.

В приложении должны быть реализованы следующие функции:

- возможность задать имя кошке в любой момент, но один единственный раз;
- кормить кошку – что увеличивает ее состояние здоровья на 1;
- наказывать кошку – что уменьшает ее состояние здоровья на 1;
- возраст кошки запрашивается продавцом при ее покупке и в дальнейшем не доступен для чтения/изменения;
- окраска кошки меняется в зависимости от состояния ее здоровья (вычисляемое свойство. Свойство возвращает значение на основании здоровья: красный окрас – здоровье < 3; желтый = 5; зеленый > 5)

Каждая Кошка, отслеживаемая системой, характеризуется следующим набором атрибутов:

- Name – write first Property
- Age – read only Property
- Heath – private Field, изменяемое функциями Feed и Punish
- Feed () – public Function
- Punish () – public Function
- Color – вычисляемое поле. Результат дает на основании свойства Heath

Функциональные требования

Идентификатор	Описание требования
FR001	Система должна позволять задать имя кошке один единственный раз
FR002	Система должна позволять кормить и наказывать кошку: <ul style="list-style-type: none">• при кормлении состояние здоровья кошки улучшается;• при наказании состояние здоровья кошки ухудшается.
FR003	Система должна позволить указать возраст кошки при покупке.
FR004	Система должна позволять узнавать информацию о состоянии здоровья кошки по ее окрасу.

Требования к интерфейсу пользователя

Идентификатор	Описание требования
UI001	Вывод информации реализуется в соответствии с представлением разработчика о проекте.

Среда выполнения, инструменты, технологии

Среда разработки: MS VS 2015/2017.

Клиент часть

.Net console-приложение.

В качестве языка реализации следует выбрать C# 6.0

Среда выполнения: Microsoft .Net Framework 4.6 или выше

Требования к реализации

Основные

1. Необходимо использовать ОО подходы при построении системы.
2. Именованые типов и переменных должно быть выдержано в единой схеме (желательно в рекомендованной Microsoft для .Net приложений, но другие также допустимы).
3. UI должен быть построен с учётом интересов пользователя. Пользоваться им должно быть удобно (а лучше и интуитивно понятно).

Дополнительные

1. Построение двухслойной системы будет дополнительным плюсом (Model + View)