

Дорогой соискатель!

Направляем вам тестовое задание на вакансию “Unity C# программист”.

Формат выполнения тестового задания:

- Задание должно быть выполнено в формате git-репозитория, размещенного на сервисе, позволяющем получить публичный доступ к репозиторию. Ссылку на репозиторий необходимо отправить в электронном письме, в теме письма указать “Выполненное задание кандидата на позицию Unity C# программист __ (Фамилия, Имя)”

Техническое задание

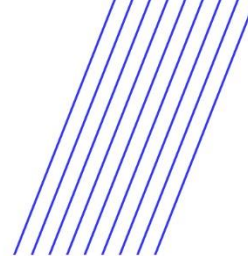
Реализовать небольшой тренажер по замене предохранителя в условном электрическом устройстве.

Тренажер должен представлять из себя приложение, написанное с помощью Unity версии не ниже 2019.3.2 на языке C# для платформы Windows. В качестве инструментов ввода необходимо использовать стандартные клавиатуру и мышь.

Из основного меню приложения можно выбрать над какой моделью (v.1 или v.2) пользователь будет производить технические работы по замене предохранителя.

Описание последовательности работ по обслуживанию модели v.1

1. Отключить напряжение, подаваемое на устройство, нажатием на кнопку выключателя.
2. Открыть задвижку верхней крышки
3. Открыть верхнюю крышку, потянув за ручку крышки
4. Отщёлкнуть задвижки, держателя предохранителя
5. Откинуть держатель предохранителя
6. Изъять из устройства вышедший из строя предохранитель
7. Положить на место новый предохранитель
8. Закрыть держатель предохранителя
9. Защёлкнуть все задвижки держателя предохранителя
10. Закрыть верхнюю крышку устройства
11. Закрыть щеколду верхней крышки
12. Подать напряжение на устройство, нажатием на кнопку выключателя.



Работа с моделью v.2 отличается только наличием под верхней крышкой устройства заслонки, скрывающей предохранитель, данная заслонка блокируется и разблокируется нажатием на кнопку на заслонке. При начале работ, необходимо сначала нажать кнопку, разблокирующую заслонку а потом отодвинуть её до упора вправо. По завершении работ необходимо сначала задвинуть заслонку, а потом заблокировать ее повторным нажатием на кнопку блокировки положения заслонки.

Реализация осуществления сценария работы с устройством версии v2 должна быть выполнена без написания дополнительного кода. То есть должны быть использованы только уже имеющиеся компоненты. Дополнительная логика должна быть внедрена за счет размещения на сцене новых объектов и изменения настроек компонентов уже имеющихся объектов.

В коде проекта должно присутствовать разделение ответственности Model View Controller либо аналоги. Важна понятная, расширяемая и поддерживаемая структура кода. Требуется отделить Model от View и Controller, объекты Model не должны наследоваться от MonoBehaviour.

В случае нарушения последовательности обслуживания устройства пользователю должно выводиться сообщение о том, что он совершил неправильное действие, после этого предложено продолжить обслуживание устройства, либо полностью перезапустить сценарий. По окончании обслуживания устройства пользователь должен увидеть статистику со временем, затраченным на обслуживание и количеством совершённых ошибок.

Внешний вид и визуальная составляющая не важны, весь тренинг может состоять из кубов разного цвета.

Оцениваемые навыки:

- Знание языка C#, принципов ООП, КОП, архитектурных подходов и паттернов проектирования.
- Навык оформления написанного кода, использование xml-документации
- Навыки работы с Unity, организация иерархии на сцене и в проекте, использование инструментов, пакетов и т.д.

Все тестовые задания должны быть выполнены вами самостоятельно и являться результатом исключительно вашей интеллектуальной деятельности, созданы без нарушения прав третьих лиц, незаконных заимствований и плагиата. Направляя результат выполненных тестовых заданий, вы подтверждаете, что понимаете и принимаете тот факт, что Крок иммерсивные технологии могло получить или получать от третьих лиц, либо разработать или разрабатывать самостоятельно материалы, схожие с направленными Вами.