

РАЗРАБОТКА ВИРТУАЛЬНОГО ПРИБОРА СИМУЛЯТОРА РОБОТОТЕХНИЧЕСКОЙ ПЛАТФОРМЫ НА ОСНОВЕ UNITY ENGINE



Автор работы

Рыжих Виктор Максимович

Барнаул 2024



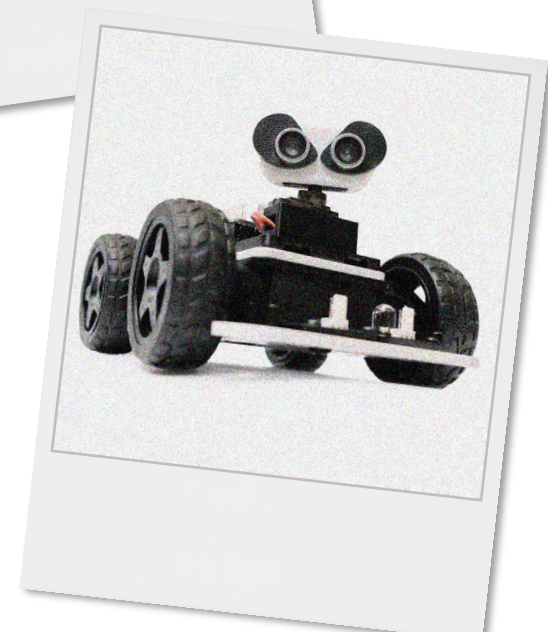
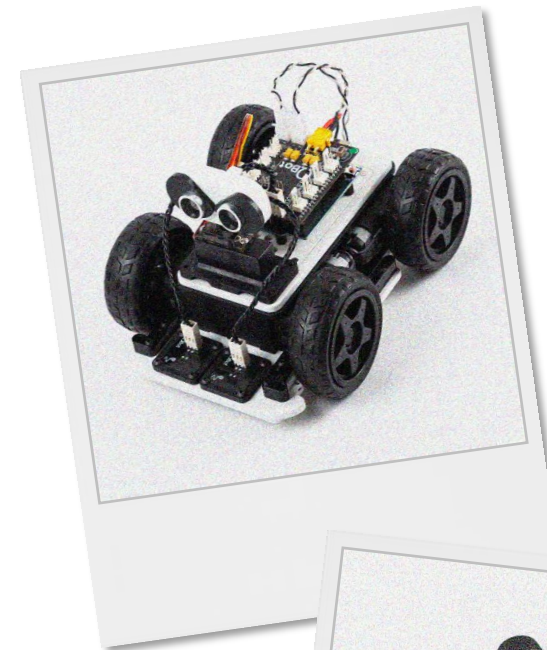
Цель разработки

Целью данной работы является создание программной среды для симуляции работы робототехнической платформы с использованием Unity Engine.

Для достижения этой цели были поставлены следующие задачи:

- выбор средств разработки;
- создание 2D-модели и интерфейса;
- разработка интерпретатора;
- интеграция в Unity.

Целью симуляции была выбрана робототехническая платформа OmegaBot от российской компании ОмегаЛаб.





Актуальность, полезность

Появляется всё больше направлений и специальностей, связанных с робототехникой, в том числе с программированием робототехнических платформ.

Разрабатываемая среда позволит с лёгкостью обучать пользователей, не знакомых с программированием микроконтроллеров, основам.

Кроме того, симулятор обеспечивает безопасное и экономически выгодное тестирование алгоритмов без необходимости использования реальных физических платформ, что особенно важно для образовательных учреждений и исследовательских лабораторий.



Средства разработки

Unity был выбран для создания симулятора из-за его гибкости и мощных инструментов для 3D-графики и физики, что позволяет точно моделировать робототехнические платформы. Платформа поддерживает кроссплатформенную разработку, обеспечивая доступность симулятора на различных устройствах. Кроме того, широкое сообщество и множество готовых ресурсов ускоряют процесс разработки и позволяют эффективно решать возникающие задачи.



Aseprite был выбран для создания визуальной модели благодаря своей специализации на пиксельной графике и простоте в использовании, что позволяет быстро и эффективно создавать детализированные и стильные спрайты для интерфейса симулятора.

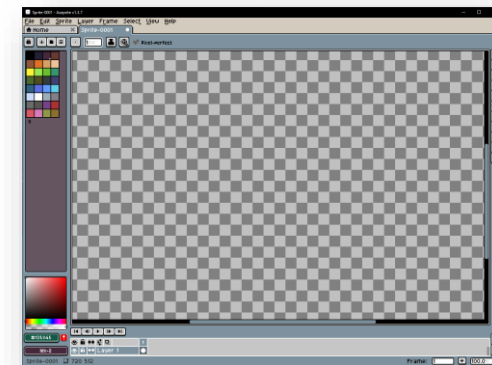
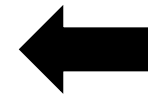
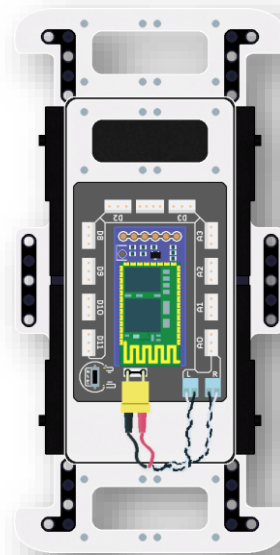
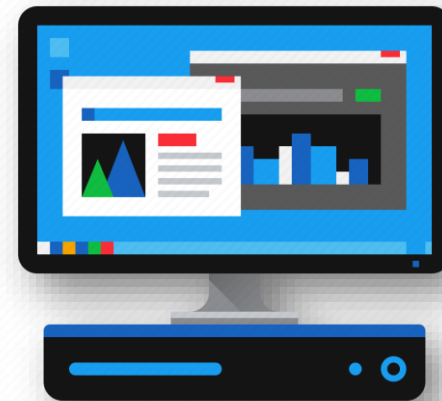


Создание модели

1. Планирование и эскиз: На первом этапе создаются наброски и концепты модели, определяются основные элементы дизайна, стиль и цветовая палитра.

2. Реализация в Aseprite: На втором этапе выполняется детальная прорисовка модели в Aseprite, включая создание всех необходимых спрайтов и анимаций, с учетом пиксельного стиля и требований к интерфейсу.

3. Тестирование и доработка: На финальном этапе модель интегрируется в симулятор, проверяется её совместимость с остальными элементами интерфейса, после чего выполняются необходимые коррекции и оптимизации.





Разработка интерпретатора

Определение грамматики языка

- Использование ANTLR (Another Tool for Language Recognition) для создания синтаксического анализатора
- Определение лексических и синтаксических правил для языка C
- Создание файла грамматики для ANTLR

Генерация парсера и лексера

- Компиляция грамматики с помощью ANTLR
- Генерация парсера и лексера на языке C#

Создание абстрактного синтаксического дерева (AST)

- Инициализация парсера с входным кодом на языке C
- Преобразование кода в AST

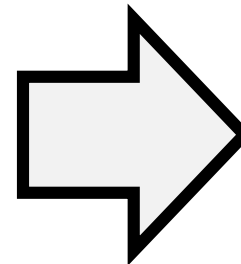
```
#include <stdio.h>

int main() {
    char name[50];

    printf("Введите ваше имя: ");
    scanf("%s", name);

    printf("Привет, %s!\n", name);

    return 0;
}
```



```
using System;

class Program
{
    static void Main() {
        Console.Write("Введите ваше имя: ");
        string name = Console.ReadLine();

        Console.WriteLine("Привет, " +
            name + "!");
    }
}
```

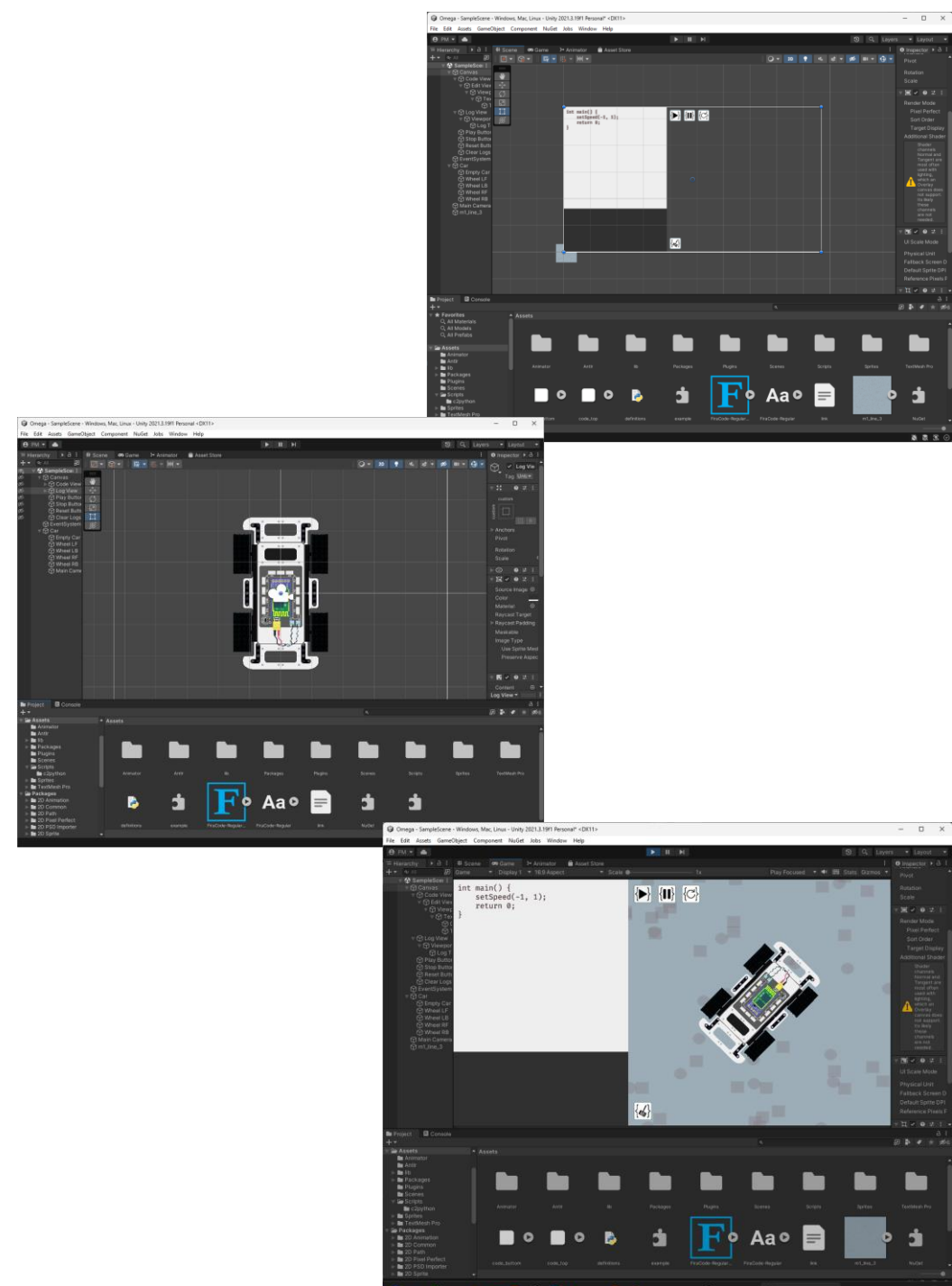


Работа в Unity

1.Создание сцены: Разработка начинается с настройки основной сцены, где размещаются все объекты, такие как модели роботов, физические препятствия и элементы окружения. Настраиваются камеры и слои сцены для оптимального отображения симуляции.

2.Программирование логики: На следующем этапе пишутся скрипты на C#, которые управляют поведением объектов, моделируют физику, обрабатывают ввод от пользователя и взаимодействие объектов в симуляции.

3.Тестирование и отладка: Завершающий этап включает запуск симуляции для проверки её корректной работы. Проводится отладка и оптимизация, чтобы гарантировать стабильность и точность физической модели, после чего проект компилируется для целевых платформ.





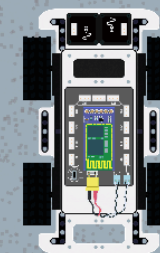
Результаты работы

Получившийся прототип симулятора представляет собой интерактивную среду, в которой можно моделировать и тестировать робототехнические платформы.

Прототип включает базовую 2D-сцену с визуальной моделью робота, созданной в пиксельном стиле, и позволяет пользователю программировать робота на языке C, а затем наблюдать за выполнением кода в реальном времени.

Симулятор поддерживает физически точную модель движения, визуализацию интерфейса и основные функции отладки, что позволяет разработчикам и учащимся тестировать алгоритмы без необходимости использования реального оборудования.

```
void setup() { }  
  
float th = 1.5;  
  
void loop() {  
    int l = getColor(1),  
        r = getColor(2);  
    bool l_black = l < 100,  
         r_black = r < 100;  
    if (l_black && !r_black)  
        setSpeed(-th, th);  
    else if (!l_black && r_black)  
        setSpeed(th, -th);  
    else setSpeed(th, th);  
}
```



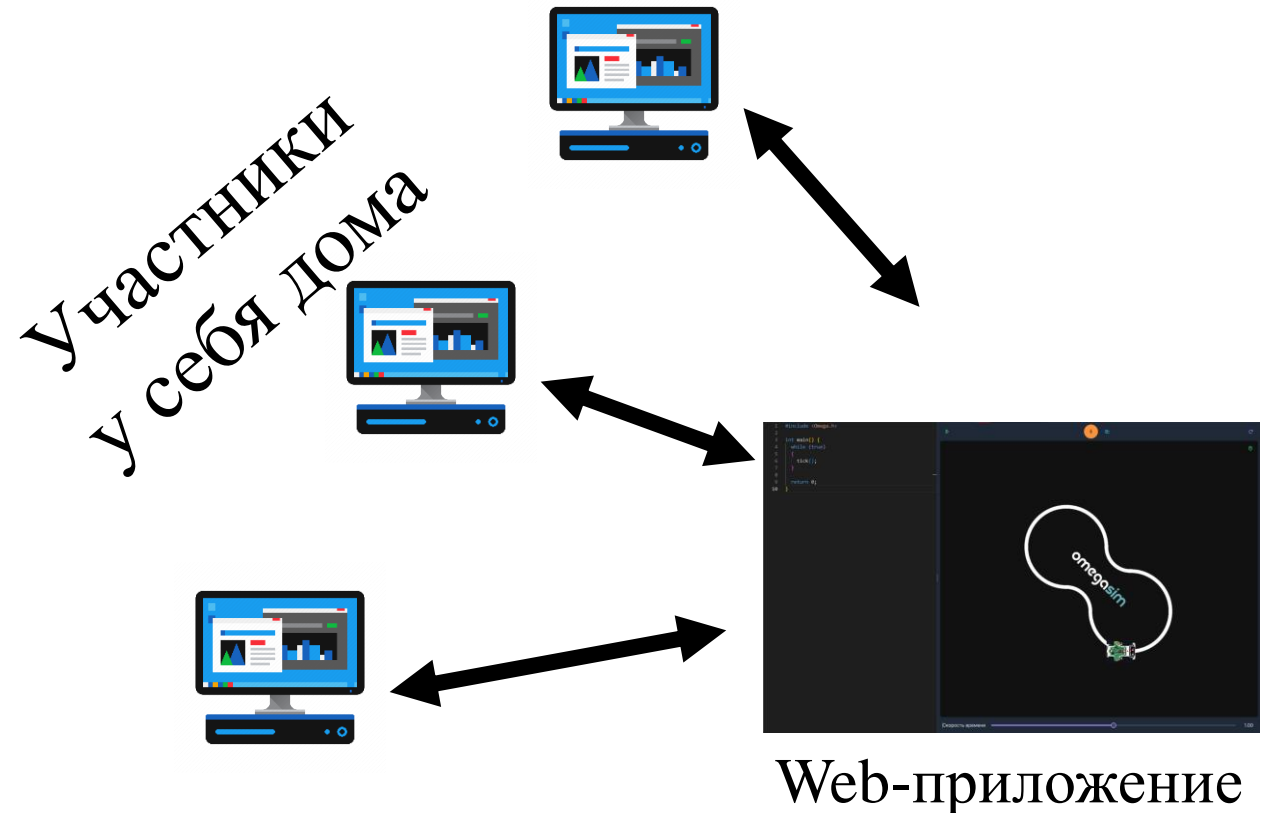


Использование наработок

Приложение-симулятор было протестировано в ходе проведения отборочного тура международной олимпиады по программированию микроконтроллеров в мае 2024 года.

Прототип симулятора был адаптирован для веб-версии с использованием Unity WebGL, что позволило запустить его в браузере без необходимости установки дополнительного ПО.

Это обеспечило доступность симулятора для участников олимпиады, которые могли программировать и тестировать свои алгоритмы прямо на веб-странице. В ходе олимпиады участники использовали веб-симулятор для выполнения практических заданий, что позволило оценить их навыки программирования и решения задач в условиях, приближенных к реальной работе с робототехническими платформами.





Перспективы развития

Симулятор спроектирован с учетом масштабируемости, что позволит расширить его функциональные возможности за счет добавления новых модулей и интеграции с внешними системами, например, внешний редактор кода или экспортирование кода в другие системы.

Симулятор масштабируется для симуляции других платформ, для чего необходима только визуальная модель и описание её физических свойств.

Это открывает возможности для создания комплексных учебных и исследовательских комплексов, где несколько систем могут работать совместно, обеспечивая интегрированный подход к обучению и тестированию робототехнических решений.