

# ПРОЕКТИРОВАНИЕ ВИРТУАЛЬНОГО ПОЛИГОНА ДЛЯ БЕСПИЛОТНОГО АВТОМОБИЛЯ

Н.И. Журбич

Томский политехнический университет

niz1@tpu.ru

## Введение

С каждым днём к нам приближается недалекое и так отчетливо видимое электронное будущее, которое принесет нам массу нововведений.

Уже сегодня мы можем наблюдать за рождением новых, ярких идей и технологий. Одной из наиболее интересных, перспективных и массовых технологий является идея создания беспилотного автотранспорта.

Статья посвящена исследованию и проектированию беспилотного автомобиля. В современном мире данное изобретение необходимо, так как на сегодняшний день человечество не может представить себя без транспорта [1].

Для проектирования данного программного комплекса необходимо построить диаграмму вариантов использования, диаграмму потоков данных и диаграмму классов.

## Обзор предметной области

«Беспилотные» автомобили, управляемые без участия водителя, разрабатываются с 1980-х годов компаниями по производству легковых автомобилей, «внутризаводского» и грузового транспорта, сельскохозяйственных машин и автомобильной техники военного назначения.

Наиболее активно работы ведутся в США, Германии, Японии, Китае, Великобритании, такими фирмами, как "General Motors", Volkswagen, "Audi" и другими. Значительный объем работ проводится по закрытой тематике в рамках оборонных заказов и по этой причине результаты работ в открытой печати не публикуются.

**Проблема** - отсутствие аналогов программного комплекса на рынке. Большинство компаний по разработке беспилотных автомобилей создают свои собственные полигоны для тестирования алгоритмов движения автомобиля. Создание таких полигонов занимает большое количество времени и требует больших затрат.

**Цель** - проектирование программного комплекса, который позволит проводить виртуальное испытание алгоритмов движения беспилотного автомобиля.

Проектирование подразумевает выработку свойств и функций системы на основе анализа предметной области.

В процессе проектирования ПО для выражения его характеристик используются различные

нотации: блок-схемы, диаграммы классов, UML-диаграммы.

## Диаграмма потоков данных

На рисунке 1 представлена контекстная диаграмма в нотации DFD.



Рис. 1. Диаграмма потоков данных в нотации UML для программного комплекса

Уровень контекстной диаграммы определяет масштаб и границы системы [2].

После составления контекстной диаграммы рассматриваемый комплекс необходимо разбить на модули, т.е. декомпозировать.

Программный комплекс будет состоять из нескольких модулей:

1. Модуль виртуальной среды (работа с 3D графикой: добавление дорог, зданий и других моделей для наполнения полигона).
2. Модуль движения автомобиля (разработка алгоритмов движения автомобиля по полигону).
3. Модуль движения пешехода (разработка алгоритмов для движения пешехода по тротуарам, пешеходным переходам и т.д.).
4. Модуль управления светофорами (управление движением автомобиля и пешехода по виртуальному полигону с помощью алгоритмов).

## Диаграмма классов

Диаграмма классов предназначена для демонстрации классов системы, их атрибутов (свойств) и связей между ними. На диаграмме классы изображены в рамках.

На 2 рисунке представлена диаграмма классов в нотации UML для проектируемого программного комплекса [3].

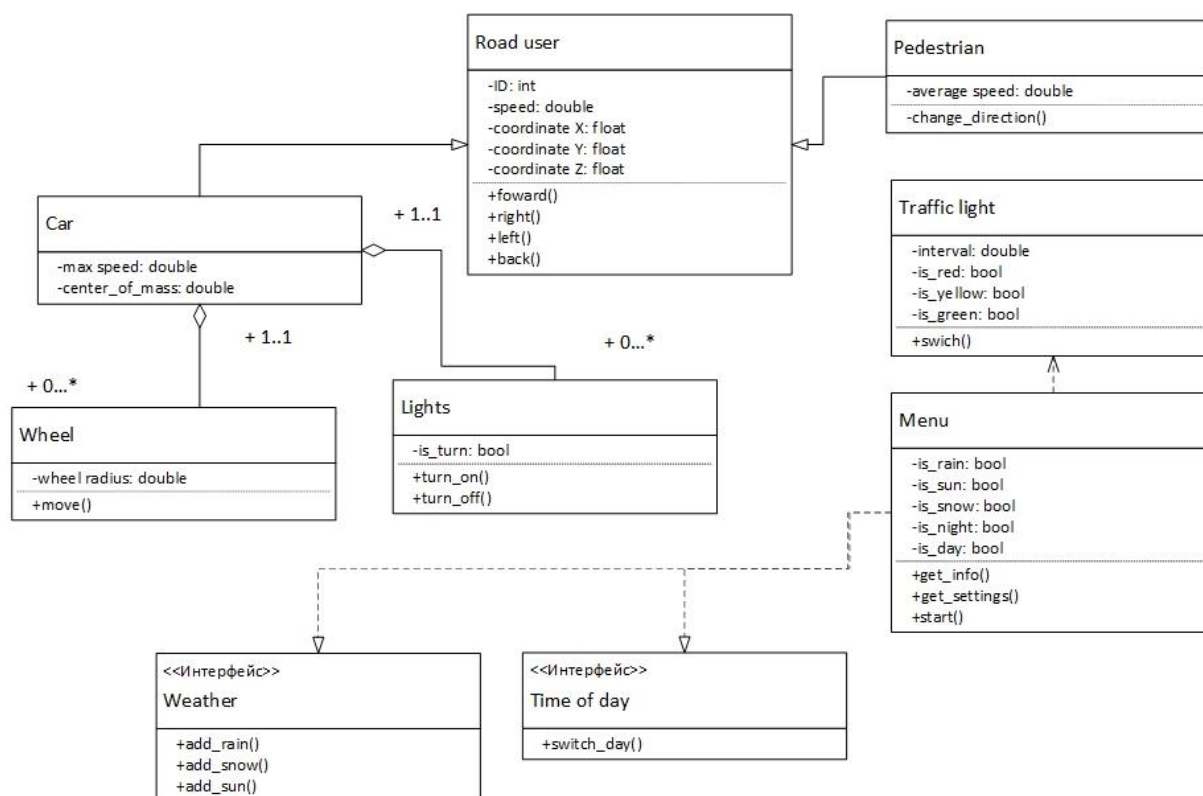


Рис. 2. Диаграмма классов в нотации UML для программного комплекса

### Выбор инструмента разработки

Для разработки данного программного комплекса была выбрана межплатформенная среда разработки компьютерных игр Unity 3D.

Unity 3D – это инструмент для разработки трехмерных приложений, работающий под любыми операционными системами.

Основными преимуществами Unity 3D являются наличие визуальной среды разработки, межплатформенной поддержки и модульной системы компонентов. К недостаткам относят работу с многокомпонентными схемами и подключением внешних библиотек.

Также одним из преимуществ данного инструмента является возможность создавать приложения для запуска в браузерах с помощью специального подключаемого модуля Unity (Unity Web Player), а также с помощью реализации технологии WebGL (Web-based Graphics Library – программная библиотека для языка программирования JavaScript).

Движок Unity 3D поддерживает два сценарных языка: C# и JavaScript (модификация). При разработке данного продукта будут использоваться оба языка программирования. Для реализации работы светофоров и других элементов полигона будет использоваться язык C#, для работы с алгоритмами движения пешехода и беспилотного автомобиля будет использоваться язык JavaScript.

### Заключение

В данной работе был спроектирован программный комплекс, позволяющий производить тестирование алгоритмов движения беспилотного автомобиля в виртуальном полигоне.

В процессе проектирования были построены следующие диаграммы:

- диаграмма классов (Static Structure diagram);
- диаграмма вариантов использования (Use-Case Diagram)
- диаграмма потоков данных и её декомпозиция (Data Flow Diagram).

### Список использованных источников

1. Япония потратит более \$25 млрд на беспилотные автомобили [Электронный ресурс]. ФАН. URL: <https://riafan.ru/789196-yaponiya-potratit-bolee> (дата обращения: 15.03.2018 г.).
2. DFD методология. [Электронный ресурс]. На заметку. URL: <https://e-educ.ru/bd14.html> (дата обращения: 01.04.2018).
3. UML-диаграммы классов. [Электронный ресурс]. UML-моделирование. URL: <https://prog-cpp.ru/uml-classes/> (дата обращения: 26.05.2018).