

ОБГРУНТУВАННЯ ДОЦІЛЬНОСТІ РОЗРОБКИ ПРОГРАМНОГО РУШІЯ ДЛЯ ВІЗУАЛІЗАЦІЇ ІНТЕРАКТИВНОЇ ГРАФІКИ

Вінницький національний технічний університет

Анотація

У роботі проаналізовано актуальність розробки програмного рушія для візуалізації інтерактивної графіки. Обґрунтовано доцільність використання фреймворка Angular та мови програмування TypeScript. Розглянуто відомі реалізації систем «рушій»

Ключові слова: рушій, Angular, TypeScript, інтерактивна графіка

Abstract

The relevance of the development of the software engine for the visualization of interactive graphics is analyzed in the paper. The expediency of using the Angular framework and the TypeScript programming language in the development is substantiated. Known implementations of "engine" systems are considered

Keywords: engine, Angular, TypeScript, interactive graphics

Вступ

HTML разом із CSS3 та JavaScript дають розробникові досить широкі можливості створення анімацій, ігор, Canvas, математики, кольорів, звуку, WebGL. Одна з найбільших переваг HTML полягає у його незалежності як від платформи, так і від апаратного забезпечення. Можна виявити ще такі додаткові можливості, що надаються рушіями: спрощення деяких задач, що часто зустрічаються, оформлене введення, фізика, звук. Є і досить слабо оформлені рушії, а є й такі, що надають у користування розробникові редактор 2D-рівнів та інструменти відлагодження. Передбачається, що більшість рушіїв служать для скорочення часових витрат на розробку повноцінної анімації. Проте багато розробників вважають за краще створювати свій проект повністю з нуля, щоби краще представляти його улаштування. Є небагато JavaScript рушіїв, які дійсно досить якісні, проте й у них може бути великий недолік: вони більше або не підтримуються або близькі до припинення такої підтримки. Тому, вибираючи рушій, найкраще зупиняти вибір на продуктах, підтримка яких триватиме певний час.

Актуальність

Не секрет, що анімація є досить популярним явищем у сучасному веб-дизайні. Вона робить зображення живим, і, відповідно, краще запам'ятовується. Як наслідок, сайти, у дизайні яких вона була застосована, мають більше шансів запам'ятатися користувачеві та привернути його увагу до себе знову. Правильно підібрана швидкість та плавність анімації, викликає у підсвідомості користувача зацікавленість і довіру до веб-ресурсу.

На даний момент існує багато різних варіацій рушіїв. Проте, враховуючи стрімкі темпи розвитку технологій, більшість з них не встигає за сучасними тенденціями і швидко стають не актуальними, потребуючи нового підходу, який би дозволив створити рушій, що буде максимально відповідати потребам завдання.

Результати дослідження

Рушій робить симуляцію фізичних законів реального світу у світі віртуальному. Найчастіше рушії використовуються не як окремі самостійні програмні продукти, а як складені компоненти інших програм. Усі рушії умовно можна поділити на два типи: ігрові та наукові. Наукові використовуються у науково-дослідних розрахунках і симуляціях, де вкрай важлива саме фізична точність обчислень. Разом із тим, швидкість обчислень не грає істотної ролі. Ігрові рушії використовуються у комп'ютерних

іграх. У цьому випадку вони повинні працювати у режимі реального часу, тобто відтворювати фізичні процеси у грі з тією же швидкістю, з якою вони відбуваються у реальному світі. Разом із тим, від ігрового рушія не потрібно високої точності обчислень: тут головна вимога – візуальна реалістичність, і для її досягнення не обов'язково проводити високоточну симуляцію. Тому в іграх використовуються досить приблизні апроксимації. Сучасні фізичні рушії симулюють не всі фізичні закони реального світу, а лише деякі, причому із часом і прогресом у галузі інформаційних технологій і обчислювальної техніки список «підтримуваних» законів збільшується.

Далі дуже стисло розглянемо кілька рушіїв.

Crafty – легкий модульний ігровий рушія, що має досить багато різних функцій: анімацію, управління подіями, перемальовування регіонів, відстежування перетинів і зіткнень, спрайтову графіку тощо. Підтримує практично всі браузері. Жодних додаткових зусиль докладати не потрібно [1].

Enchant.js – фреймворк Enchant.js для HTML5+JavaScript ігор був розроблений у 2011 році, поширюється з відкритим вихідним кодом (MIT ліцензія) і тому є безкоштовний [2].

GameJS – велика бібліотека на верхньому рівні HTML Canvas. Окрім функцій малювання у ній є збільшуваний набір корисних для розробки ігор модулів. Більшість наявного API ґрунтується на популярній PyGame [3].

Cocos2D – портований з iPhone графічний 2D HTML5 рушія на базі JavaScript. Дозволяє швидко створювати 2D-ігри і графічні додатки, що можуть працювати на всіх сучасних пристроях без встановлення додаткових плагінів [4].

Phaser – фреймворк для створення мобільних і комп'ютерних ігор на HTML5 із застосуванням Canvas і WebGL. Безкоштовний і з відкритим вихідним кодом. Є довідники для швидкого старту роботи на JavaScript і TypeScript [5].

Three.js – це бібліотека JavaScript та інтерфейс прикладного програмування (API), що використовується для створення та відображення анімованої 3D-комп'ютерної графіки у веб-браузері за допомогою WebGL. Вихідний код розміщений у сховищі на GitHub [6]. Three.js дозволяє створювати 3D-анімацію, прискорену графічним процесором (GPU), використовуючи мову JavaScript як частину веб-сайту, не покладаючись на власні плагіни браузера. Це можливо завдяки появі WebGL. Бібліотеки високого рівня, такі як Three.js або GLGE, SceneJS, PhiloGL або ряд інших бібліотек, дозволяють створювати складні 3D-комп'ютерні анімації, які відображаються у браузері, без зусиль, необхідних для традиційного автономного додатка або плагіна.

TypeScript – це об'єктно-орієнтована мова програмування, розроблена корпорацією Microsoft. Це мова з відкритим вихідним кодом, призначена для створення великомасштабних веб-програм, а JavaScript – це мова програмування на стороні сервера, що дозволяє розробляти інтерактивні веб-сторінки. Хоча на виході компілятор створює, фактично, JavaScript, який далі виконується браузером. Проте сувора типізація мінімізує кількість потенційних помилок, які могли б виникнути під час розробки на JavaScript [7].

JavaScript – динамічна, об'єктно-орієнтована (прототипна) мова програмування [8]. Найчастіше використовується для створення сценаріїв вебсторінок, що надає можливість на боці клієнта (пристрої кінцевого користувача) взаємодіяти з користувачем, керувати браузером, асинхронно обмінюватися даними з сервером, змінювати структуру та зовнішній вигляд вебсторінки. JavaScript має низку властивостей об'єктно-орієнтованої мови, але завдяки концепції прототипів, підтримка об'єктів у ній відрізняється від традиційних мов ООП. Крім того, JavaScript має ряд властивостей, притаманних функціональним мовам: функції як об'єкти першого рівня, об'єкти як списки, анонімні функції, замикання що додає мові додаткову гнучкість.

Angular – досить популярний веб-фреймворк, написаний на безпечному типі JavaScript (TypeScript), який використовує деякі функції та основні бібліотеки, які можуть імпортувати користувацькі програми [9]. Використовуючи Angular з такими можливостями TypeScript, як служби, ін'єкція залежностей і маршрутизація, розробник може створити інтерфейс, схожий на рідний API. У результаті отримуємо програму, яка працює як на iOS, так і на Android. Angular це складна платформа, яку важко освоїти, тому вона вимагає належної кваліфікації від розробника. Численні структурні елементи, які містять інжектори, компоненти, директиви, послуги тощо, можуть бути складними для початківців розробників для вивчення. Однак вони є великим плюсом для успіху продукту, оскільки можна створити все, що тільки придумаєте, за допомогою повного набору вбудованих в Angular функцій.

Висновки

Обґрунтовано актуальність розробки програмного рушія для візуалізації інтерактивної графіки. Розглянуто деякі популярні програмні рушії та проаналізовано їх переваги та недоліки. Обґрунтовано доцільність використання фреймворку Angular та мови програмування TypeScript.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Crafty URL: <https://www.npmjs.com/package/craftyjs> (дата звернення 22.05.2022)
2. Enchant.js URL: <https://www.npmjs.com/package/enchant> (дата звернення 22.05.2022)
3. GameJs URL: <https://www.npmjs.com/package/gamejs> (дата звернення 22.05.2022)
4. Cocos2d URL: <https://ru.wikipedia.org/wiki/Cocos2d> (дата звернення 22.05.2022)
5. Phaser URL: <https://phaser.io/> (дата звернення 22.05.2022)
6. Threejs URL: <https://threejs.org/> (дата звернення 22.05.2022)
7. JavaScript URL: <https://uk.wikipedia.org/wiki/JavaScript> (дата звернення 22.05.2022)
8. Мова програмування URL: <https://www.ua5.org/osnprog/220-mova-programuvannja.html> (дата звернення 22.05.2022)
9. Angular URL: <https://angular.io/> (дата звернення 22.05.2022)

Максименюк Максим Русланович – студент групи 2КН-186, факультет інтелектуальних інформаційних технологій та автоматизації, Вінницький національний технічний університет, м. Вінниця, e-mail: maksymmaksymeniuk@gmail.com

Арсенюк Ігор Ростиславович – доцент кафедри комп'ютерних наук, Вінницький національний технічний університет

Maksymeniuk R. Maksym – student of group 2CS-18b, Faculty of Intelligent Information Technology and Automation, Vinnytsia National Technical University, Vinnytsia, e-mail: maksymmaksymeniuk@gmail.com

Igor R. Arsenyuk – Cand. Sc., Assistant Professor of the Chair of Computer Science, Vinnytsia National Technical University, Vinnytsia