Performance Comparison of Augmented Reality Frameworks

Dafnis Cain Villagran-Vizcarra
dept. de Industria y Manufactura, Instituto de ingeniería y Tecnología
Universidad Autónoma de Ciudad Juárez
Chihuahua, Mexico
al206595@alumnos.uacj.mx

David Luviano-Cruz
dept. de Industria y Manufactura, Instituto de ingeniería y Tecnología
Universidad Autónoma de Ciudad Juárez
Chihuahua, Mexico
david.luviano@uacj.mx

Luis Asunción Pérez-Domínguez
dept. de Industria y Manufactura, Instituto de ingeniería y Tecnología
Universidad Autónoma de Ciudad Juárez
Chihuahua, Mexico
luis.dominguez@uacj.mx

Abstract— **In our quest to incorporate Augmented Reality (AR) into industrial and university laboratories for training purposes, we conducted an analysis of four AR frameworks. Our goal was to develop a portable Starter Kit (SK) and determine the most sustainable option for our project. This kit integrates both hardware and software components, designed to improve an optimize AR functionality on computers, smartphones, and tablets. The research involves four key stages: beginning with NAS configuration, followed by 3D model creation, next is the generation of QR code identifiers, and finally, the development of a cross-platform (C-P) solution.**

Keywords—Augmented reality, augmented reality frameworks, Mobile and Cross-platform

I. INTRODUCTION

AR involves blending digital elements into the physical environment through screens, creating immersive simulations [1]. AR integration into industrial training faces challenges, such as network limitations [2]. This study aims to address these challenges by developing cost-effective AR solutions that can contribute to the wider adoption of AR in the industrial sector.

II. THEORETICAL AND CONCEPTUAL FRAMEWORKS

This section focuses on implementing Augmented Reality (AR) using Android and iOS, with iOS encompassing both iOS and the recent iPadOS [3]. iOS finds versatility in tourist centers and 3D learning environments, with iPads preferred for educational applications [4]. On the other hand, Android, an open-source operating system developed by Google, is widely used in various contexts, including children's reading with AR, video game development, training in reinforced concrete usage, and practice simulators for educational purposes [1]. Vuforia AR, a software development kit (SDK) with Unity, excels in image recognition, text rendering, direction identification, and target tracking, with applications in museums and realistic AR interactions [5]. ARJS, a JavaScript(JS) based framework, simplifies AR development, enhances performance, and accommodates older devices, used in molecular visualization, modeling, and dynamic web content development [6]. Finally, Google's <model-viewer> (MV) is a specialized tag that aids in understanding various components, including AR modeling [7].

III. METHODOLOGY

A. Software and Hardware Configuration

This section covers the essential software and hardware configuration. Android® and iOS® platforms were considered for Augmented Reality (AR) integration, with a requirement of ARCore® certification. Specific mobile phone and tablet models suitable for this project include Motorola, Samsung, Asus, Google, Nokia, and Xiaomi [7].

The selected mobile device is the Xiaomi 10T 5G smartphone, while AR prototypes are developed using a Lenovo Laptop Idea Pad L340 [3]. Unity® is the primary software tool for AR development, with four foundational AR frameworks: ARJS®, EasyAR®, Vuforia®, and Google <Model-Viewer>®.

Vuforia® excels in image recognition and text rendering for applications like museums [5]. EasyAR® is known for its user-friendly interface. ARJS® streamlines development, enables older devices to operate at 60 frames per second, and spans from molecular visualization to dynamic web content [6]. Google <Model-Viewer>® is a proprietary Google framework focused on AR modeling [7].

The Glitch® platform is used for implementing <Model-Viewer> and ARJS® frameworks, and Blender is crucial for creating AR 3D models. Evaluation of AR development across these frameworks involves the participation of an average user for valuable feedback on the user experience.

B. Methodology

The research starts with an AR software literature review for selecting user-friendly AR frameworks, proceeds to a comprehensive analysis of framework webpages for guidance, and then involves creating, testing, and analyzing four prototypes, with a focus on resource utilization.

Active involvement of an average user in tasks such as AR app development and website creation, documented with video recordings, provides valuable insights.

The elements to measure are programming language (L), response time, resources used, additional software, software, application, and platform.

C. Prototype Development Methodology

The prototype development process describes series of steps, with some important requisites of 3D models crafted in Blender, producing file extensions "GBL," "USDZ," "FBX," and "GLTF." The operational course includes distinct phases, each tailored to the unique attributes of the employed AR tools.

1) Google <Model-Viewer> Framework

It begins by visiting the <Model-Viewer> page and selecting the Glitch® hyperlink, leading to preliminary prototype design. "Remix This" is chosen to access code manipulation, and 3D models in "GLB" and "USDZ" formats are added to the "assets" folder.

2) ARJS Framework

"AR.js studio", an open-source AR platform, is employed with a location-based approach, using latitude and longitude coordinates. This processl involves uploading 3D models in "GLB" format and integrating them into an HTML document via the Glitch® platform.

3) EasyAR Framework

EasyAR requires creating an account and downloading the "EasyAR Sense Unity" plugin. Unity® is used with specific components, including "Android® SDK & NDK Tools" and "OpenJDK." The configuration involves company details and image "Target Data."

4) Vuforia Framework

This framework needs a "Vuforia®" account, followed by the download of the "Vuforia Engine SDK and Import Manually" package. The development includes database creation, target data addition, and 3D model integration. This methodology ensures the systematic progression from model generation to prototype development, navigating the distinct attributes of each AR framework with precision.

The methodology consists of seven steps: framework selection, documentation review, prototype development, application or server execution, response time measurement, memory usage monitoring, and data collection. This ensures a systematic progression from model generation to prototype development, tailored to the unique characteristics of each augmented reality framework.

IV. RESULTS

The results of this study are presented in Table 1; however, other findings indicate that Easy AFrame ARJS offers a viable approach to creating applications using HTML files and "glb" format 3D models with image identifiers. Android devices face challenges related to camera permissions, which necessitate the use of Model-Viewer. Model-Viewer streamlines project creation and management on the glitch.com platform, allowing access to "glb" files online. While the Easy AR framework facilitates straightforward prototype development within the Unity environment, the management of 3D modeling files may become complex. Vuforia is another viable option, easy to implement but requiring account verification before utilization.

TABLE. 1. SPECIFICATIONS FOR MEASURING PERFORMANCE, FRAMEWORKS CHARACTERISTICS AND DEVELOPMENT REQUIREMENT.

	Framework performance		Frameworks characteristics		Development requirement
	Time of answer	*Resource used*	*L*	*C-P*	*Application*
ARJS	0.7sec	800KB	JS	*	-
MV	1sec	100KB	JS	*	-
Easy-AR	0.2sec	21.5KB	C#	app mobile	23.2MB
Vuforia-AR	0.1sec	26.1KB	C#	app mobile	28.1MB

V. CONCLUSIONS

AR tools streamline development, but evaluating practical features in each framework is crucial. Contemporary AR development is feasible with these tools, fostering research. Identifying user-friendly, adaptable, and fast frameworks is key. These findings highlight AR tools' potential to enhance diverse sectors, driving adoption and innovation. The future of AR promises innovation and progress.

REFERENCES

[1] J. Miguel *et al.*, «La formación de ingenieros en sistemas automotrices mediante la realidad aumentada», *Innovación educativa*, vol. 20, p. 25, jul. 2020.

[2] T. Braud, F. H. Bijarbooneh, D. Chatzopoulos, y P. Hui, «Future Networking Challenges: The Case of Mobile Augmented Reality», *Proceedings - International Conference on Distributed Computing Systems*, n.º June, pp. 1796-1807, 2017, doi: 10.1109/ICDCS.2017.48.

[3] Rocío García, «Qué es iOS: Características y versiones del sistema operativo de Apple». febrero de 2021. Accedido: 30 de mayo de 2021. [En línea]. Disponible en: https://www.adslzone.net/reportajes/software/que-es-ios/

[4] M. Callejas, J. Quiroga, y A. Alarcón, «Ambiente Interactivo Para Visualizar Sitios Turísticos, Mediante Realidad Aumentada Implementando Layar», *Ciencia E Ingeniería Neogranadina*, vol. 21, n.º 2, pp. 91-106, 2011.

[5] Taban, «Introducción a Vuforia (Realidad aumentada) – Taban». mayo de 2018. Accedido: 30 de mayo de 2021. [En línea]. Disponible en: http://taban.mx/2018/05/16/introduccion-a-vuforia-realidad-aumentada/

[6] M. Ramírez, «AR.js acerca la realidad aumentada a la web de forma gratuita | Creativos Online».

[7] Google, «Etiquetas especiales que Google entiende | Centro de la Búsqueda».