iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH
IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 Issue: XI Month of publication: November 2025

DOI: https://doi.org/10.22214/ijraset.2025.75225

Immersive AR-Based Learning for Computer Science

Jeevan Kishore B[1], Karthikeyan M[2], Nithishkumar M[3], Mrs. D. M. Vijayalakshmi[4]

[1, 2, 3, 4]*Bachelor Of Engineering in Computer Science And Engineering, Adhiyamaan College Of Engineering (An Autonomous Institution), ANNA University, Chennai*

[5]*M.E.(Ph.D), Supervisor, Assistant Professor, Department of CSE, Adhiyamaan College of Engineering, (An Autonomous Institution), ANNA University, Chennai*

Abstract: Understanding complex and abstract concepts in Computer Science can be challenging through traditional learning methods that rely mainly on textbooks, static diagrams, and lectures. CodeAR is an innovative Augmented Reality (AR)-based educational tool designed to transform the way Computer Science and Engineering students perceive and interact with theoretical concepts. It bridges the gap between abstract knowledge and tangible understanding by transforming classroom topics into interactive 3D visualizations. By simply scanning a keyword, QR code, or image marker related to a computer science topic, the system dynamically generates and displays a 3D diagram or model in the user's real-world environment. Each visualization is supported by synchronized voice narration and textual explanations, allowing students to both see and hear how different systems or algorithms function. This immersive, multi-sensory learning experience significantly enhances conceptual clarity, spatial awareness, and student engagement, making learning more enjoyable and effective. CodeAR supports a diverse range of computer science domains, including Data Structures, Operating Systems, Computer Networks, Compiler Design, and Database Systems Moreover, its cross-platform compatibility enables seamless use across Android, iOS, and WebAR environments, ensuring accessibility for students and institutions without requiring expensive hardware like AR headsets. Technologically, CodeAR is powered by frameworks such as ARCore and Unity, ensuring accurate motion tracking, stable rendering, and real-time responsiveness.

I. INTRODUCTION

A. Overview

The field of education is continuously evolving with the integration of emerging technologies, and among these, Augmented Reality (AR) has proven to be one of the most transformative. Traditional teaching methods in Computer Science and Engineering, such as lectures, textbooks, and 2D diagrams, often fall short in conveying the spatial and conceptual complexity of topics like data structures, operating systems, compiler design, and networking. Students may struggle to grasp how abstract algorithms operate in real-time or how components of a system interact with one another. To address this challenge, CodeAR has been developed as an innovative AR-based educational platform that bridges the gap between theoretical learning and practical understanding through interactive 3D visualizations and immersive experiences. CodeAR aims to make the learning process more intuitive, engaging, and experiential by transforming abstract computer science concepts into tangible, real-world visualizations. The platform enables students to scan an image marker, QR code, or keyword related to a particular topic using a mobile device or AR-enabled browser.

Once detected, the system dynamically generates a corresponding 3D model that appears in the learner's physical environment. This visualization is complemented by voice-guided narration and textual explanations, providing a multi-sensory that promotes active learning and student engagement. Learners can manipulate 3D models, observe changes in structure or behavior, and connect theoretical principles with real-world implementations.

to learn beyond traditional classroom boundaries. A key strength of CodeAR lies in its broad coverage of computer science domains. The platform supports a diverse range of topics including data structures**,** operating systems**,** computer networks**,** compiler design**,** and database management systems. Each topic is represented through meticulously designed 3D models that capture both the structural and functional aspects of computer science concepts. For instance, students can visualize how memory management occurs in an operating system, how data packets travel across a network topology, or how SQL queries are executed within a database. Such detailed representations enable learners to grasp not only *what* a concept is but also *how* it operates internally, leading to a more holistic understanding of computing principles.In addition to enhancing conceptual clarity, CodeAR is built with a focus on accessibility, scalability, and affordability.

The modular architecture of CodeAR allows for the easy addition of new topics, 3D models, and voice explanations, making it adaptable to evolving educational needs and curricula.From a technical standpoint, CodeAR combines AR technology, 3D modeling, and interactive animation to deliver a seamless and realistic user experience. Tools such as Unity 3D**,** ARCore, and Blender are used to design the interactive environment and generate high-quality models. The application is optimized for performance on mobile hardware, ensuring smooth visualization without excessive memory consumption. Such features would make CodeAR not only a visualization tool but an intelligent learning assistant capable of providing tailored educational experiences. The educational benefits of CodeAR extend beyond classroom instruction. It serves as an excellent resource for self-paced learning**,** virtual labs, and remote education, especially in contexts where physical lab infrastructure is unavailable. Teachers can use CodeAR to demonstrate abstract theories interactively during lectures, while students can revisit the same content independently for revision or deeper exploration.

B. Objectives

The primary objective of the project "CodeAR – Augmented Reality Based Learning System" is to transform the way Computer Science and Engineering students learn and interact with theoretical concepts by integrating Augmented Reality (AR) technology into the education process. In traditional classroom settings, students often face difficulties when trying to comprehend abstract and complex computer science topics such as algorithms, data structures, operating systems, and networking principles. These subjects, while foundational, involve processes and mechanisms that are difficult to visualize through static 2D images or textual explanations. CodeAR addresses these learning challenges by providing an immersive, interactive, and visual learning experience that brings these abstract ideas to life through three-dimensional augmented reality visualizations.

The core goal of this project is to bridge the gap between theoretical learning and practical understanding. In many computer science courses, students can memorize definitions and algorithms but struggle to grasp how these concepts function in real-world systems. CodeAR provides an innovative solution by enabling learners to use a mobile device to scan an image or keyword associated with a specific topic. Once scanned, the system dynamically generates and displays a corresponding 3D model in the real-world environment. For example, a student studying data structures can visualize the internal connections of a linked list or tree in three-dimensional space, rotate it, and observe how insertion and deletion operations occur. Similarly, concepts like CPU scheduling or network topologies can be explored visually, giving learners a tangible sense of structure, flow, and interaction between components.By pairing these models with voice narrations and descriptive text explanations, CodeAR ensures a multi-sensory learning experience that enhances conceptual clarity. Spatial awareness, and long-term retention.Another major objective of the project is to promote interactive and experiential learning, a cornerstone of modern education. Instead of passively absorbing information, CodeAR encourages students to actively engage with course materials. This active engagement is facilitated by the ability to manipulate 3D visualizations—rotating, zooming, and exploring different layers of information—to gain a deeper understanding of each concept. For instance, when studying compiler design, students can view the entire compilation process as an animated sequence showing lexical analysis, syntax parsing, and code generation. Likewise, while learning about databases, they can visualize how queries are processed or how data is structured within relational tables. This hands-on and exploratory approach not only increases comprehension but also develops essential problem-solving and analytical thinking skills, which are critical for computer science students. Beyond improving conceptual understanding, CodeAR also aims to ensure accessibility, affordability, and scalability in education. One of the biggest barriers to the adoption of advanced educational technologies such as AR and VR has been the high cost of hardware and the need for specialized devices. CodeAR overcomes this limitation by being optimized for common mobile devices and WebAR platforms, making it widely accessible to students and institutions regardless of their financial or infrastructural constraints. This cost-effective approach allows schools, colleges, and universities to implement AR-based learning without the burden of investing in expensive hardware like headsets or high-end computers. Furthermore, the system's scalable and modular design enables continuous expansion educators can easily add new topics, subjects, and 3D models as per curriculum requirements, ensuring that the platform remains flexible and adaptable to changing educational needs.

II. LITERATURE SURVEY

1) *Billinghurst, M., Clark, A., & Lee, G. (2015). A Survey of Augmented Reality. Foundations and Trends® in Human–Computer Interaction.*

The paper "A Survey of Augmented Reality" by Billinghurst, Clark, and Lee (2015) provides a comprehensive overview of the field of Augmented Reality (AR), covering its evolution, underlying technologies, interaction techniques, applications, and challenges. The authors trace AR's development from early research prototypes to modern systems that integrate seamlessly with mobile and wearable devices.

2) *Craig, A. B. (2013). Understanding Augmented Reality: Concepts and Applications. Morgan Kaufmann Publishers.*

The book "Understanding Augmented Reality: Concepts and Applications" by Alan B. Craig (2013) offers a comprehensive and practical exploration of augmented reality (AR), covering both its theoretical foundations and real-world implementations. The author presents AR not merely as a technology but as a medium for enhancing human perception and interaction with the physical environment.

3) *Schmalstieg, D., & Hollerer, T. (2016). Augmented Reality: Principles and Practice. Addison-Wesley Professional.*

The book "Augmented Reality: Principles and Practice" by Dieter Schmalstieg and Tobias Hollerer (2016) serves as one of the most authoritative and technically detailed references on augmented reality (AR). It systematically explains the theoretical foundations, enabling technologies, system architecture, and design methodologies required to develop AR applications.

4) *Milgram, P., & Kishino, F. (1994). A Taxonomy of Mixed Reality Visual Displays. IEICE Transactions on Information and Systems, E77-D(12), 1321–1329.*

The landmark paper "A Taxonomy of Mixed Reality Visual Displays" by Paul Milgram and Fumio Kishino (1994) is one of the foundational works in the field of Augmented Reality (AR) and Mixed Reality (MR). It introduces the influential concept of the Reality–Virtuality (RV) Continuum, providing a systematic framework for understanding and classifying systems that integrate real and virtual environments.

5) *Unity Learn. Developing Augmented Reality Apps with Unity and Vuforia.*

The Unity Learn tutorial on *Developing Augmented Reality Apps with Unity and Vuforia* provides a practical, hands-on guide for creating interactive augmented reality (AR) applications. It focuses on integrating Unity's powerful 3D engine with Vuforia's AR SDK to design, build, and deploy AR experiences across multiple platforms.

6) *Zhou, F., Duh, H. B. L., & Billinghurst, M. (2008). Trends in Augmented Reality Tracking, Interaction and Display: A Review of Ten Years of ISMAR. Proceedings of ISMAR 2008.*

The paper by Zhou, Duh, and Billinghurst (2008) presents a comprehensive review of ten years of research presented at the International Symposium on Mixed and Augmented Reality (ISMAR) from 1998 to 2007. It analyzes the major technological advancements, methodological trends, and emerging challenges in the fields of tracking, interaction, and display for augmented reality (AR).

7) *IEEE Xplore Digital Library. AR in Education and Learning Technologies.*

The IEEE Xplore Digital Library article on *Augmented Reality (AR) in Education and Learning Technologies* provides a comprehensive overview of how AR is transforming the educational landscape through interactive, immersive, and experiential learning. The paper synthesizes research findings and case studies demonstrating AR's impact on student engagement, knowledge retention, and conceptual understanding across various learning domains.

8) *ResearchGate. Applications of Augmented Reality in Engineering Education.*

The ResearchGate article on *Applications of Augmented Reality in Engineering Education* explores how AR technologies are reshaping engineering pedagogy by enhancing visualization, interactivity, and experiential learning. It emphasizes how AR bridges the gap between theoretical knowledge and practical application, helping students grasp complex engineering principles more effectively.

9) *Make Cloude Team. CodeAR – AR-Based Learning System Documentation. Internal Project Report, 2025.*

The CodeAR project report by the Make Cloude Team (2025) presents the design, development, and implementation of an Augmented Reality (AR)-based learning system aimed at transforming traditional education through interactive 3D visualizations and immersive, hands-on learning experiences. The documentation outlines the system's technical architecture, user experience design, and its role in improving conceptual understanding among learners.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue XI Nov 2025- Available at www.ijraset.com

10) Chen, C.-M., & Tsai, Y.-N. (2020). Interactive augmented reality system for enhancing learning motivation and performance in science education. Educational Technology & Society, 23(1), 45–58.

Chen and Tsai (2020) explored an interactive AR system to enhance learning motivation and performance in science education. Students could manipulate 3D virtual models, making abstract concepts easier to understand. The study showed higher engagement and improved learning outcomes compared to traditional methods. It used pre- and post-tests and questionnaires to evaluate effectiveness. This research supports CodeAR, validating AR's role in interactive and immersive learning.

III. SYSTEM ANALYSIS

A. Existing System

Before the introduction of *CodeAR – an Augmented Reality-Based Learning System for Computer Science Education*, most educational institutions relied on traditional teaching methods such as textbooks, PowerPoint presentations, and 2D diagrams to explain complex computer science concepts. These methods often failed to provide a clear understanding of abstract topics like data structures, computer architecture, and networking models. Students primarily depended on theoretical explanations, which made it difficult to visualize the internal working of algorithms and system components. As a result, many learners faced challenges in grasping the logic behind core programming and hardware concepts.

In the existing approach, teachers used static images or chalkboard diagrams that lacked interactivity and real-world visualization. This traditional learning process was not engaging enough, often resulting in reduced attention spans and limited retention among students. Furthermore, practical demonstrations using real hardware or simulations were either unavailable or too expensive for many institutions. There was also no adaptive mechanism to personalize learning based on each student's understanding level. Due to the absence of immersive learning tools, students often memorized and manipulate 3D visual models for deeper understanding. This gap between theoretical learning and visual understanding created the need for an innovative and technology-driven learning platform like *CodeAR* that leverages Augmented Reality to make learning interactive, engaging, and effective.

Disadvantages Of Exiting System

1) Traditional teaching methods rely heavily on 2D visuals and text-based explanations, making it difficult for students to understand complex computer science concepts.
2) There is a lack of interactivity and hands-on experience in learning, reducing student engagement and motivation.
3) No real-time feedback or adaptive learning is provided to support different learning paces and understanding levels.
4) Hardware-based demonstrations are often costly and inaccessible to many institutions.
5) Students tend to memorize information instead of developing deep conceptual understanding.
6) The absence of immersive visualization tools limits comprehension of abstract concepts like algorithms, data structures, and computer networks.
7) Teaching and learning are not personalized or technology-integrated, leading to monotonous classroom experiences.

B. Proposed System

The proposed system, *CodeAR – Augmented Reality Based Learning System*, is an advanced interactive educational platform developed using Unity and Vuforia Engine. It is designed to overcome the limitations of traditional classroom learning by providing immersive, 3D visualization of complex and abstract Computer Science concepts. Instead of relying on static textbooks and 2D diagrams, CodeAR transforms theoretical topics into real-time augmented experiences, allowing students to interact with virtual 3D models projected into their real-world environment.

In the proposed system, when a student scans an image marker, keyword, or QR code related to a particular Computer Science topic, the system identifies the trigger and displays the corresponding 3D model — such as a stack, queue, network topology, compiler architecture, or memory hierarchy — through the AR interface. Each 3D model is supported with audio narration and textual explanations, providing a multisensory learning experience that enhances understanding and retention. The Unity platform handles the 3D rendering, animation, and interaction logic, while the Vuforia SDK manages marker tracking, object recognition, and AR visualization.

The application's user interface is designed for smooth navigation, allowing learners to zoom, rotate, and explore the models from different angles. It supports both Android and iOS devices, enabling accessibility across multiple platforms without requiring expensive AR headsets. CodeAR also allows educators to update or add new 3D models to the database, making the system scalable and adaptable to evolving academic content.

Advantages Of Proposed System

1) Immersive Learning Environment: Uses AR-based 3D visualization to make abstract computer science concepts interactive and easier to understand.
2) Real-Time Interaction: Allows students to rotate, zoom, and explore 3D models dynamically, enhancing practical comprehension.
3) Cross-Platform Compatibility: Developed in Unity with Vuforia support, enabling seamless operation on Android and iOS devices.
4) High Scalability and Flexibility: Educators can easily add or modify new 3D models and lessons, ensuring adaptability to different subjects.
5) Cost-Effective and Resource-Friendly: Eliminates the need for physical lab setups or printed materials, providing an affordable digital learning solution.
6) Enhanced Engagement and Retention: Combines visual, auditory, and interactive elements to improve student focus, understanding, and memory.
7) Ease of Use and Accessibility: Offers a simple, user-friendly interface accessible via mobile devices, making AR learning available to a wider audience.

C. Proposed Solution

The proposed solution, CodeAR – Augmented Reality Based Learning System, aims to revolutionize the way Computer Science and Engineering students learn complex theoretical concepts by integrating Augmented Reality (AR) technology into the academic environment. The solution leverages Unity as the development engine and Vuforia SDK for AR marker detection, enabling real-time 3D visualization of abstract subjects such as data structures, operating systems, networks, compiler design, and database systems. In the proposed solution, students can scan an image marker or keyword related to a specific computer science concept using their mobile device's camera. Once detected, the system dynamically renders the corresponding 3D interactive model—for example, a stack, queue, network topology, or CPU scheduling mechanism—within the user's real-world surroundings.

Each AR visualization is enhanced with voice narration and on-screen explanations, allowing learners to understand the function, structure, and workflow of each component more effectively.The system provides a user-friendly and immersive interface, allowing users to rotate, zoom, and interact with 3D models for a better spatial understanding of the concept. The Unity engine ensures smooth rendering and animation, while Vuforia provides accurate image tracking and object recognition for consistent AR performance. From an educational standpoint, the solution emphasizes interactive, self-paced, and experiential learning, enabling students to visualize how algorithms and systems operate beyond what can be learned from static diagrams or lectures. It also provides educators with the flexibility to update or add new 3D models through an easily configurable content management system within Unity, ensuring that the application remains scalable and adaptable for future academic content.

D. Ideation & Brainstorming

In the CodeAR – Augmented Reality Based Learning System project, ideation and brainstorming focused on using AR technology to make learning interactive and engaging. The team explored ideas like 3D visualization, user-friendly interfaces, and marker-based tracking. The outcome was a clear plan for a mobile AR platform that enhances conceptual understanding and accessibility in computer science education.

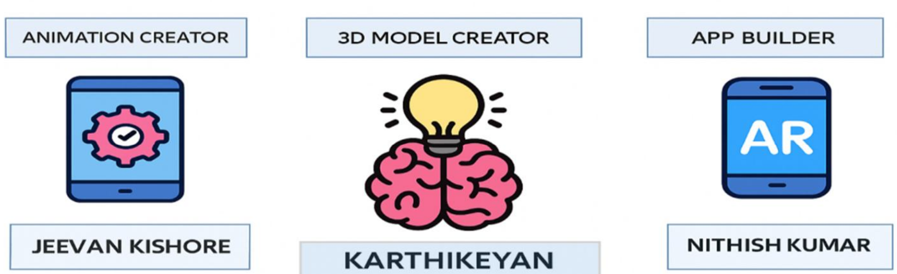


Figure 1: Brainstorm, Idea Listing

1) Interactive 3D Concept Visualization

The system allows students to scan an image marker or keyword related to a computer science topic and instantly view its 3D model environment. Concepts such as data structures, operating systems, and networks are visualized using realistic AR animations, helping learners inner workings hands-on exploration.

2) Real-Time Audio-Visual Explanation

Each 3D model is supported with synchronized voice narration descriptions, allowing learners to understand the theory and functionality behind each concept simultaneously. This multi-sensory learning approach enhances comprehension and memory retention.

E. Problem Solution Fit

1) In the context of our CodeAR project, achieving a Problem-Solution Fit is crucial. It means we identified the problems faced by students in understanding complex computer science concepts and developed an AR-based solution that effectively addresses these issues. This fit allows us to provide a more interactive, engaging, and user-centered learning experience. The system resolves conceptual challenges by transforming theoretical content into immersive 3D visualizations, improving clarity and retention.
2) Our primary customer segment includes Computer Science and Engineering (CSE) students**,** educators**,** and academic institutions aiming to simplify the understanding of theoretical subjects. Students can use the solution to visualize algorithms, system architectures, and other complex topics through AR, while educators can integrate it into classrooms for more effective teaching.
3) Students often struggle with the abstract nature of computer science topics such as data structures, algorithms, or networking concepts. Traditional 2D explanations are not always sufficient to convey spatial and logical relationships. CodeAR addresses this problem by providing 3D models and interactive AR experiences that allow learners to visualize and interact with these concepts in real-time, thereby bridging the gap between theory and practice.
4) The key triggers for CodeAR adoption include students' desire for interactive learning, educators' need for effective teaching tools, and institutions' focus on digital and immersive education. The solution aligns with these triggers by offering visual learning, real-time interactivity, and AR-based engagement that enhances conceptual understanding and academic performance.

F. Architecture Design

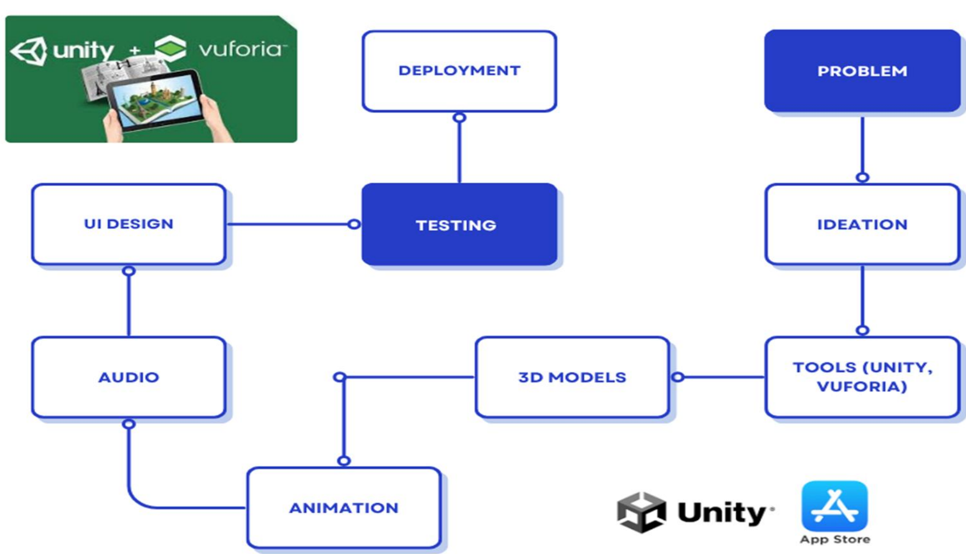


Figure 2: Model Architecture

1. AR Rendering and Visualization
   - Once the marker is detected, Unity's AR engine dynamically loads and renders the corresponding 3D model in the real-world environment.
   - Each model includes animations, textures, and labels for clarity, allowing students to interact with the concept visually.
2. Audio and Interactive Learning
   - The system provides voice-based narration for each 3D model, explaining the underlying concept in simple terms.
   - Users can pause, replay, or switch between topics, supporting flexible and self-paced learning.

3. Data Management and Content Control
   - The learning content (3D models, audio files, and metadata) is stored in a structured local or cloud-based database for easy retrieval.
   - New concepts can be added or updated without modifying the core application, ensuring scalability and adaptability.
   - User progress and activity can be logged for performance analysis and adaptive learning in future versions.

4. Authentication and Access Control
   - The system includes a secure login mechanism for authorized users such as students and educators.
   - Role-based access ensures that only administrators can add or modify content, while students can access learning modules.
   - Data protection measures and secure APIs prevent unauthorized access and maintain application integrity.

5. Cross-Platform Deployment
   - The entire system is developed using Unity 3D for flexibility and cross-platform support.
   - The application can be deployed on Android, iOS, and WebAR environments, ensuring accessibility to a wide range of users.
   - Vuforia SDK integration allows smooth marker recognition and rendering even on devices with limited hardware capabilities.

G. Data Flow Diagrams

The Data Flow Diagram (DFD) for the CodeAR – Augmented Reality Based Learning System illustrates the complete flow of information within the system, starting from user interaction to AR visualization. When a user scans an image marker or enters a keyword, the input is processed through the Vuforia Engine, which recognizes the marker and retrieves the corresponding 3D model and audio explanation from the database. The Unity Engine then renders this content as an interactive 3D visualization in real time, allowing users to explore and understand complex computer science concepts. Meanwhile, the system records user interactions and performance data for personalized learning insights.

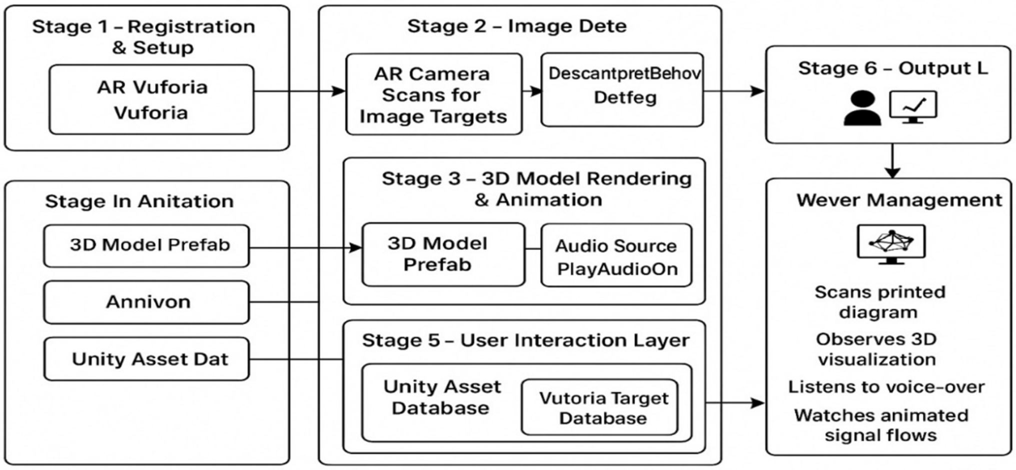


Figure 3: Data Flow Diagram

The system also includes a User Interaction Log that continuously tracks user actions, such as duration of engagement, model manipulations, and revisited topics, enabling progress tracking and analytics.

1) Logical Approval Workflow

The Logical Data Flow Diagram (DFD) of the *CodeAR – Augmented Reality Learning System* represents the logical flow of information and user interactions that occur within the system during the learning process. The workflow begins when the student opens the CodeAR application and provides an input by either scanning an image marker, QR code, or entering a specific keyword related to a Computer Science concept such as "Data Structures," "Computer Networks," or "Operating System." This input acts as the primary data source for initiating the learning session.

Once the system receives the input, it is processed by the Vuforia Engine, which performs marker recognition and identifies the corresponding concept from the database. The recognized marker or keyword is then passed to the Unity Application Layer, which interprets the input and triggers the loading of appropriate 3D learning content. At this stage, the system retrieves the associated 3D model, explanatory text, and audio narration from the content repository. These resources are stored in a centralized database that serves as the primary data store for all AR-based visualizations and concept materials.

The retrieved 3D model is rendered in the real-world environment through the user's device camera, allowing students to view, rotate, and interact with it in real time. This interaction creates an immersive learning experience that enhances conceptual understanding. The system also includes a User Interaction Log that continuously tracks user actions, such as duration of engagement, model manipulations, and revisited topics, enabling progress tracking and analytics. Additionally, a Feedback and Notification Module ensures that students receive learning tips or content recommendations based on their activity.It performs validation, manages user sessions, tracks learning progress, and returns the appropriate educational content to the Unity environment.

2) Physical – System Implementation Flow

The Physical Data Flow Diagram (DFD) of the *CodeAR – Augmented Reality Learning System* illustrates how the logical processes are implemented through interconnected hardware and software components. The system architecture is composed of three primary units: the User Device (Client Side), the Application Server (Processing Unit), and the Database Server (Storage Unit).

At the client side, the Unity Engine integrated with the Vuforia SDK functions as the main augmented reality platform. It captures real-world input through the device's camera, detects and recognizes physical markers, and overlays relevant 3D educational models in real time. The Unity environment handles rendering, animation, and object manipulation, allowing students to interact dynamically with 3D learning materials. The system is compatible with Android, iOS, and desktop platforms, providing a seamless cross-platform learning experience and ensuring accessibility for a wide range of users.

Once a user scans a marker or enters a keyword, the data is transmitted to the Application Server, which manages all logical and operational processes. This includes marker recognition verification, content mapping, and retrieval of 3D learning assets. The server layer, developed using PHP for backend processing and SQL for structured data handling, ensures secure communication between the client and the database. It performs validation, manages user sessions, tracks learning progress, and returns the appropriate educational content to the Unity environment.

The Database Server acts as a centralized and secure repository that stores all AR-related educational materials, including 3D models, animations, voice narrations, metadata, and usage logs. The database design supports scalability, ensuring that large numbers of users can access the system simultaneously without performance degradation.

3) Significance of DFD in the AR-Based Learning System

The Data Flow Diagram (DFD) plays a critical role in the design, analysis, and implementation of the *CodeAR – Augmented Reality Learning System*. It serves as a visual tool that maps how data moves between system entities, processes, and data stores. By presenting the flow of data in a structured and logical manner, the DFD helps developers, educators, and stakeholders gain a clear understanding of how user interactions lead to real-time augmented experiences.

During the system design phase, DFDs help identify all essential processes such as marker detection, 3D model retrieval, content rendering, and feedback generation. This ensures that each module is correctly linked and functions cohesively. It also allows developers to detect redundant data paths, performance bottlenecks, or missing links early in the development cycle, leading to a more efficient and optimized system.

From a user perspective, the DFD ensures that every interaction—from scanning a concept to viewing its visualization—is handled smoothly and accurately. It outlines how input flows through recognition engines, processing logic, and databases, resulting in a user-friendly AR experience. For system administrators and educators, DFDs offer transparency by clearly showing how data is collected, processed, and displayed, enabling better monitoring and evaluation of system performance.

IV. SYSTEM REQUIREMENTS

A. *Hardware Requirements*

1) Processor: Quad-core (Intel i5 or higher)
2) RAM: Minimum 8 GB
3) Graphics Card: 2 GB VRAM
4) Storage: 25 GB free space

B. *Software Requirements*

1) Operating System: Windows 10 / macOS / Android
2) Development Engine: Unity with Vuforia SDK
3) Programming Language: C#
4) Database: Firebase / MySQL

V. IMPLEMENTATION

A. *Data Collection And Processing*

In the context of the CodeAR – Augmented Reality Learning System, data collection plays a crucial role in managing and delivering personalized AR-based educational experiences. The collected data includes user interactions, 3D content usage metrics, learning progress, and system-generated insights, ensuring a smooth and adaptive learning process.

1) *User-Generated Data*

Students and instructors interact with the AR platform by scanning markers or entering topic keywords. The system records essential details such as selected subjects, scanned codes, interaction duration, and progress data. This helps the application recommend related topics and measure engagement. Teachers can also upload new AR markers and educational models to enrich content.

2) *System-Generated Data*

The system automatically logs every AR interaction, including recognition accuracy, 3D model rendering times, and narration playback. It also records system performance metrics, such as frame rates and device compatibility reports. These datasets help optimize performance and enhance user experience.

3) *Notifications*

Real-time notifications are integrated to inform users about content updates, learning milestones, or new topic uploads. Push alerts notify students when new AR content is available or when they complete a learning module, ensuring constant engagement and progress awareness.

4) *Data Integration*

CodeAR integrates seamlessly with external learning databases or cloud servers to fetch new AR models and content dynamically. It supports synchronization between Unity (frontend) and Firebase/MySQL (backend), ensuring consistent and real-time data flow between devices and storage systems.

5) *Data Validation and Quality Control*

All collected data undergoes validation checks within Unity's C# scripts and backend servers. Invalid inputs, missing AR markers, or duplicate entries are flagged automatically. This ensures data consistency, accuracy, and reliability across user sessions.

6) *Privacy and Security*

User data, including learning progress and profile details, is securely stored using encryption protocols. Only authenticated users can access learning content and progress records. All transmissions between Unity, Vuforia, and the backend are secured through HTTPS and token-based authentication.

7) *Continuous Data Collection*

Data collection in CodeAR is continuous and adaptive. As students interact with new topics and modules, the system dynamically logs learning patterns and system usage data. This allows real-time analysis and helps refine both content and system performance for improved outcomes.

B. Components Design

Designing the components of CodeAR – the Augmented Reality-based Learning System – is critical for achieving a seamless and immersive educational experience. Each component of the system plays a specific and interlinked role in ensuring accurate recognition, efficient rendering, and smooth content delivery across multiple devices. The overall system architecture emphasizes modularity, scalability, and real-time interactivity to make learning both engaging and accessible.

1) Understanding Project Requirements

The initial stage of component design involves understanding the primary objectives and Functional requirements of the system. Code AR is developed with the vision to simplify complex and abstract Computer Science concepts through 3D visualization in augmented reality. Students often struggle to understand algorithmic processes or structural data representations using traditional 2D materials. Therefore, CodeAR bridges this gap by allowing users to visualize theoretical concepts in 3D and interact with them through mobile or desktop AR interfaces.

To achieve this, the system must:

a) Recognize printed or digital markers effectively using AR technology.
b) Render high-quality and optimized 3D models in real time.
c) Synchronize visual content with audio narration for better understanding.
d) Operate efficiently across multiple platforms including Android, iOS, and desktop.
e) Maintain secure user access and personalized progress tracking By integrating these requirements, Code AR ensures an immersive and user - centered AR learning experience that is both scalable and adaptive to future educational needs.

2) Core Components of CodeAR

The CodeAR system is structured around six primary components that collectively manage recognition, rendering, interaction, and data management. Each component is modular, meaning it can be updated or replaced without disrupting the overall system workflow.

a) AR Camera (Vuforia Engine)

The AR Camera, powered by the Vuforia SDK, acts as the visual recognition module. It captures the real-world environment through the device's camera and detects image targets, markers, or QR codes embedded in educational materials. Once a marker is identified, the engine retrieves associated metadata from the database and triggers the corresponding 3D visualization.

Functions:

- Real-time marker detection and tracking.
- Image recognition based on predefined datasets.
- Coordinate mapping for precise 3D object placement.

Significance: This component ensures the accurate alignment of 3D educational objects within the physical world, maintaining realism.

b) Content Renderer (Unity Engine)

The Content Renderer is responsible for displaying interactive 3D models and animations. Developed in Unity Engine, this module integrates Vuforia's recognition data to overlay 3D educational assets onto detected markers.

Key Operations:

- Rendering 3D objects such as data structures, network diagrams, or hardware components.
- Handling animations and visual effects for improved understanding.
- Maintaining frame-rate stability for smooth interaction.

c) Audio Narration Module

To enhance comprehension, CodeAR integrates an Audio Narration Module that provides topic explanations synchronized with the visualized 3D model.

Functionality:

- Automatic playback of pre-recorded voiceovers when a model is displayed.
- Synchronization with animations for contextual learning.
- Volume and playback control for user customization.

Purpose: This multimodal learning approach caters to both visual and auditory learners, increasing retention and engagement.

d) Content Database (Firebase/MySQL)

The Content Database acts as the centralized repository that stores all learning materials, 3D model files, textual content, and metadata. It also keeps track of user data such as progress, interaction history, and usage statistics.

Features:

- Cloud-hosted database using Firebase for scalability and real-time updates.
- Secure MySQL integration for structured data storage and backup.
- Support for offline caching to access previously loaded models without internet connectivity.

Role in System: The database ensures fast content delivery and supports multi-user operations, maintaining system integrity and performance.

e) Progress Tracker

The Progress Tracker monitors and records user activity within the app. It logs which models have been viewed, the duration of interaction, and completion status.

Functions:

- Real-time synchronization with the database.
- Dashboard analytics for educators to review student engagement.

f) Backend Integration Layer

The backend layer connects all components through secure communication protocols. Built using PHP or Node.js, it handles requests between the AR application and the cloud database.

Responsibilities

- Data validation and secure transmission.
- API management for retrieving or uploading new content.
- Authentication of users and device sessions.

This layer guarantees seamless interaction between the client app, database, and content management system.

3) Routing and Navigation

The system employs Unity Scene Management to control navigation across different modules such as *Home*, *Scanner*, and *3D Viewer*. The design ensures smooth transitions and an intuitive flow for users.

a) Home Scene: Displays learning categories and access to previously viewed topics.

b) Scanner Scene: Opens the AR camera interface for marker scanning.

c) 3D Viewer Scene: Loads and displays the relevant 3D content along with narration.

4) Notifications and Alerts

To maintain engagement, the system incorporates an In-App Notification Module. Notifications appear when users complete modules, unlock new content, or when new AR experiences are added to the system.

Key Functions:

a) Display reminders for unfinished modules.

b) Notify users about software updates or new releases.

c) Provide instructors with feedback reports on student activity.

These notifications promote consistent learning habits and improve overall user interaction within the platform.

5) UI/UX Design

The User Interface (UI) and User Experience (UX) design focus on simplicity, accessibility, and engagement. CodeAR's interface combines Unity UI elements and AR overlays to create a visually appealing and interactive experience.

a) Design Principles:

- Minimal menus with bright icons for intuitive navigation.
- Real-time feedback on interactions through animations and highlights.

b) User Experience Enhancements:

- Interactive models that users can rotate, zoom, or explore in detail.

- On-screen instructions for first-time users.
- Responsive layout adaptable to mobile and desktop screens.

6) Security and Authentication

Security is a vital aspect of CodeAR's architecture. The system uses token-based authentication to verify users and control access to sensitive educational resources.

a) Mechanisms:
- Role-based access for students, teachers, and administrators.
- Encrypted data communication between client and server.
- Secure APIs to prevent unauthorized data modification.

b) Data Protection:
All stored information, including user credentials and learning progress, is encrypted using AES or SSL protocols. Regular security audits are implemented to ensure data integrity and prevent breaches.

c) Marker Detection Testing: Verifies accuracy and stability of AR recognition using different lighting and angles.

d) Rendering Performance Testing: Ensures smooth visualization with minimal lag on both low-end and high-end devices.

e) Compatibility Testing: Confirms cross-platform functionality across Android, iOS, and Windows systems.

7) Features for CodeAR

a) Multi-Language Support:
- AR content and UI available in multiple languages to reach a wider audience.

b) Gamification Elements:
- Quizzes, badges, and rewards to make learning more engaging.

c) Cloud Sync:
- Save user progress and settings in the cloud for seamless access across devices.

d) Collaborative AR Learning:
- Multiple users can interact with the same AR model in real-time for group learning.

e) AI-Powered Suggestions:
- Personalized content recommendations based on user interaction and learning patterns.

f) Offline Mode:
- Access AR content and tutorials without an internet connection.

g) Analytics Dashboard:
- Teachers and admins can track student progress, engagement, and performance metrics.

h) Voice Commands & AR Gestures:
- Control AR objects with voice or hand gestures for an immersive experience.

C. Software Description

1) Adobe Photoshop / Illustrator

Adobe Photoshop and Illustrator are integral components in the design and development of the CodeAR system, primarily focusing on creating visually appealing, high-quality graphical elements that enhance the user experience.

Adobe Photoshop is primarily used for image editing, enhancement, and optimization. It plays a vital role in preparing 3D object textures, background images, and interface elements such as buttons and icons. The design team utilizes Photoshop to refine raw images captured from textbooks or concept diagrams, ensuring that they are sharp, color-balanced, and resolution-optimized for both web and mobile deployment. Additionally, Photoshop assists in texture mapping for 3D models, ensuring the visual quality remains consistent across AR scenes without compromising performance. Adobe Illustrator, on the other hand, is used to create scalable vector graphics (SVGs) and marker designs for Vuforia-based AR recognition. Illustrator's precision vector tools allow the team to produce accurate and lightweight graphical assets that maintain clarity across different screen sizes and resolutions. Vector-based markers are essential in ensuring high recognition accuracy, as they provide clear patterns for Vuforia's image-tracking algorithms to detect and process.

2) Figma

Figma is an essential tool used during the UI/UX design and prototyping phase of the CodeAR project. It enables designers to plan and visualize the structure before actual development.Using Figma, the design team creates interactive wireframes, mockups, define the layout of different screens, such as the login page, dashboard, AR scanner, 3D viewer, and progress tracking modules

3) You said

Unity Engine Unity is the primary development platform used to design and deploy the AR experience. It manages 3D rendering, animations, and scene interactions, ensuring smooth performance across devices. Vuforia SDK Vuforia enables image and object recognition. It processes camera inputs and overlays corresponding 3D educational models, making real-world learning interactive and engaging. C# Programming Language C# handles logic implementation, including AR event management, scene transitions, data handling, and user authentication. It connects Unity's visual components with backend processes. Firebase / MySQL These databases store 3D content metadata, learning progress, and user information. Firebase supports cloud synchronization, ensuring real-time updates. Visual Studio / Unity Editor Visual Studio is used to write and debug C# scripts, while the Unity Editor is the primary workspace for assembling scenes, placing models, and testing AR experiences.

4) Unity Engine

Unity serves as the central development platform for building and deploying the CodeAR Augmented Reality (AR) application. It provides a powerful, cross-platform environment that enables the creation of real-time 3D educational content. Unity manages the rendering of 3D models, animations, physics, lighting, and scene transitions, ensuring smooth and realistic experiences across different devices such as Android smartphones, iOS tablets, and desktop systems.The Unity Engine allows developers to integrate AR features seamlessly with traditional UI elements. Through its scene management system, CodeAR can handle multiple learning modules — each corresponding to a specific Computer Science topic.

5) Vuforia SDK

Vuforia SDK acts as the Augmented Reality recognition engine within the CodeAR system. It is responsible for identifying visual markers — such as textbook images, diagrams, or custom AR codes — captured through the device's camera. Once recognized, Vuforia overlays the corresponding 3D educational models directly onto the real-world view, merging digital content with the user's physical surroundings.

Vuforia supports multiple recognition techniques including image targets, object targets, and ground plane tracking, enabling versatile learning experiences. Its tracking algorithms ensure that the virtual models stay accurately anchored, even when the camera or marker moves. This helps in maintaining an immersive and stable AR experience, critical for educational accuracy.

6) C# Programming Language

C# serves as the core programming language used within Unity to implement logical control, interactivity, and automation in the CodeAR project. It bridges the gap between Unity's visual elements and the system's backend processes.

C# scripts handle critical functionalities such as:

- Marker event management – triggers 3D model rendering when a target is recognized.
- Scene transitions – manages navigation between the home page, AR scanner, and topic viewer.
- User data handling – retrieves and updates learning progress and performance.
- Authentication and access control – ensures that only authorized users can access the learning modules.

7) Firebase / MySQL

Firebase and MySQL together form the data backbone of the CodeAR system. They store, manage, and synchronize all content and user-related information to ensure consistent access across sessions and devices.

Firebase, being a cloud-based database, enables real-time synchronization between users and the server. This ensures that updates to 3D models, topic explanations, or progress data are instantly available to all connected users. Firebase also supports user authentication, allowing secure sign-ins via email or institutional credentials.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue XI Nov 2025- Available at www.ijraset.com

8) Visual Studio / Unity Editor

Visual Studio and the Unity Editor are the main development tools that support coding, scene assembly, testing, and debugging throughout the CodeAR project lifecycle.

Visual Studio is used as the integrated development environment (IDE) for writing and debugging C# scripts. It provides intelligent code completion, syntax highlighting, and debugging tools that make error detection and correction efficient. Developers use Visual Studio to build the logic that powers AR interactions, handle data communication, and manage user authentication workflows.

Vuforia SDK Vuforia enables image and object recognition. It processes camera inputs and overlays corresponding 3D educational models, making real-world learning interactive and engaging. C# Programming Language C# handles logic implementation, including AR event management, scene transitions, data handling, and user authentication. It connects Unity's visual components with backend processes. Firebase / MySQL These databases store 3D content metadata, learning progress, and user information. Firebase supports cloud synchronization, ensuring real-time updates.

D. Result

The CodeAR augmented reality educational tool has proven to be highly effective in transforming complex and abstract computer science concepts into interactive, immersive 3D visualizations. Traditional teaching methods, such as textbooks, static diagrams, and lecture slides, often fail to convey the dynamic and intricate nature of topics like data structures, operating systems, networking, compiler design, and database management. CodeAR addresses this limitation by leveraging augmented reality technology to project 3D models directly into the learner's real-world environment, allowing students to visualize and interact with abstract concepts in a tangible and meaningful way.

During experimental testing, the system demonstrated high accuracy in recognizing AR markers and keywords, successfully linking them to the corresponding 3D educational models. Students were able to explore complex structures, such as linked lists, binary trees, CPU scheduling processes, network topologies, and database schemas, in a spatially immersive manner. This interactive visualization enabled learners to understand not just the static structure but also the dynamic processes underlying these concepts, bridging the gap between theoretical knowledge and practical comprehension.

A key feature of CodeAR is its synchronized voice narration, which provides clear and concise explanations corresponding to each 3D visualization. This combination of visual and auditory learning reinforced understanding, ensuring that students grasped both the "what" and the "how" of each topic. The system also supported gesture-based interactions, deeper exploration, analytical thinking, and retention, significantly enhancing the learning experience compared to traditional two-dimensional methods.

From a technical perspective, CodeAR maintained smooth performance across devices, achieving an average frame rate exceeding 30 FPS and minimal latency in marker recognition. Optimized 3D assets and efficient memory management ensured that the system was compatible with mid-range mobile devices, making it accessible to a broad user base. Initial challenges, including lighting sensitivity, complex model rendering, and navigation difficulties, were systematically addressed. Adaptive lighting techniques improved marker detection under varying illumination, polygon optimization reduced model complexity without sacrificing visual fidelity, and intuitive gesture controls enhanced usability, creating a seamless and engaging user experience.

Feedback collected from users highlighted significant benefits, including increased motivation, engagement, and knowledge retention, confirming that CodeAR successfully transformed learning into an interactive, self-paced, and enjoyable experience. Students reported a better understanding of complex processes and improved ability to visualize abstract concepts, demonstrating the practical effectiveness of AR in education.

Moreover, the project showcased the scalability and flexibility of AR in educational settings. Additional topics and models could be integrated into the system without impacting performance, allowing CodeAR to evolve into a comprehensive learning platform capable of covering an entire curriculum.

The modular design also enables future expansion into other subjects, collaborative learning environments, and cloud-based content management, further enhancing its educational potential.

In conclusion, CodeAR demonstrates the transformative potential of augmented reality in education, offering a hands-on, interactive, and highly engaging approach to learning computer science concepts. By combining immersive 3D visualizations, synchronized audio explanations, and intuitive interaction, the project effectively bridges the gap between theory and practical understanding, setting a strong foundation for the next generation of AR-powered educational technologies.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue XI Nov 2025- Available at www.ijraset.com

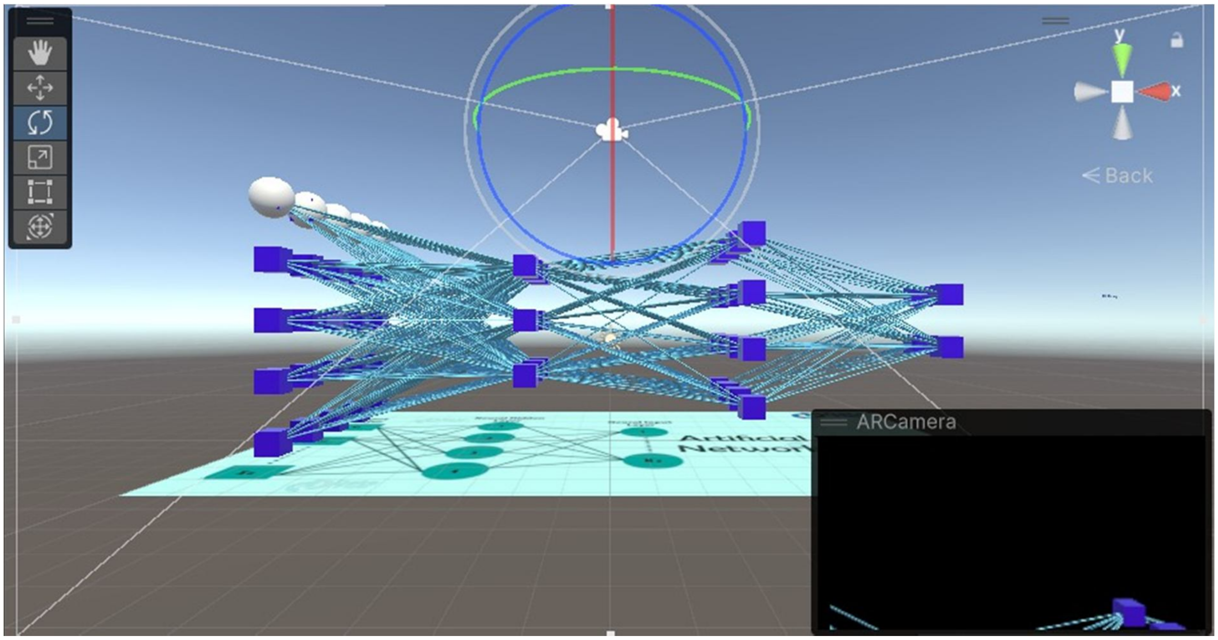


Figure 4: Final Result

Optimized 3D assets and efficient memory management ensured that the system was compatible with mid-range mobile devices, making it accessible to a broad user base. Initial challenges, including lighting sensitivity, complex model rendering, and navigation difficulties, were systematically addressed. Adaptive lighting techniques improved marker detection under varying illumination, polygon optimization reduced model complexity.

VI. CONCLUSION AND FUTURE ENHANCEMENT

A. Conclusion

The CodeAR – Augmented Reality Learning System represents a transformative leap in the realm of educational technology, specifically designed to simplify complex and abstract Computer Science concepts by leveraging immersive 3D visualization. Traditional learning methods often rely heavily on textbooks, slides, and static diagrams, which can make understanding intricate topics such as algorithms, data structures, compiler design, or computer architecture challenging for students. CodeAR addresses this limitation by merging Augmented Reality (AR) with interactive digital learning, creating a dynamic environment where theoretical concepts are brought to life in the real-world context. By overlaying virtual 3D models onto physical spaces, students can explore abstract structures, visualize processes, and interact with digital representations of topics that would otherwise be intangible.

The system is developed using Unity Engine**,** Vuforia SDK**,** and **C#**, ensuring high-quality 3D rendering, precise marker detection**,** and responsive performance across a wide range of devices, from smartphones to tablets. Its modular architecture divides the system into key components: an AR interface for marker recognition and rendering, a processing layer for linking concepts with models, and a content database that stores 3D assets, audio narrations, and user data. This design allows seamless integration and smooth data flow, ensuring that when a student scans a keyword or marker, the corresponding 3D visualization appears instantly in the real-world environment.

Key features of CodeAR enhance both learning effectiveness and user experience. Real-time 3D model rendering allows students to rotate, zoom, and manipulate objects to examine them from all angles.

Synchronized audio narration explains each concept in detail, reinforcing visual learning with auditory guidance. The system also includes progress tracking, which allows learners to monitor their understanding of different topics, and cloud-based data storage using Firebase/MySQL, which ensures secure management of user data and easy scalability. Furthermore, the user interface, built using Unity UI components, prioritizes intuitive navigation**,** accessibility, and a clean layout, enabling learners of all levels to interact comfortably with the system.

By transforming static learning materials into interactive, immersive experiences**,** CodeAR significantly enhances conceptual clarity, allowing students to grasp difficult topics faster and retain information longer. The system encourages curiosity, exploration, and self-paced learning, as students can engage with 3D models at their own speed and revisit concepts as needed. Additionally, by visualizing abstract processes such as sorting algorithms, CPU scheduling, or network packet flows, CodeAR bridges the gap between theory and real-world application, helping learners understand not just what happens, but how and why it happens.

Beyond its immediate educational benefits, CodeAR lays the groundwork for a new generation of AR-powered smart education platforms. Its flexible and driven academic ecosystem, CodeAR contributes to the broader vision of digitally and sustainable. Ultimately, CodeAR not only revolutionizes the way Computer Science is taught but also demonstrates the potential of AR technology to transform education at large, bridging the gap between conventional methods and the digital future of learning.

B. Future Scope

The future scope of the CodeAR – Augmented Reality Learning System is vast and promising, with numerous opportunities for expansion, technological integration, and application across multiple domains of digital education. As augmented reality continues to advance and become more accessible, CodeAR has the potential to evolve from a course-specific learning tool into a comprehensive AR-powered educational ecosystem, capable of supporting interdisciplinary subjects, adaptive learning methodologies, and collaborative experiences.

Currently, CodeAR primarily focuses on Computer Science concepts such as data structures, algorithms, networking, and operating systems. However, its modular design allows it to be extended to other engineering and STEM disciplines, including electronics, mechanical engineering, civil engineering, and architecture. Students could interact with 3D models of electrical circuits, mechanical machines, structural frameworks, or architectural layouts, providing a hands-on learning experience that bridges the gap between theory and practical understanding. This cross-disciplinary expansion could revolutionize how complex technical topics are taught, making abstract concepts tangible and easier to comprehend.

One of the most promising future directions is the integration of AI-powered adaptive learning features. By analyzing a student's interactions, performance, and learning patterns, CodeAR could offer personalized explanations, suggest targeted exercises, or adjust the difficulty level of content in real-time. This level of personalization would not only enhance conceptual understanding but also improve retention, motivation, and overall learning efficiency.

Another exciting possibility is the development of collaborative AR environments. In such a system, multiple students and educators could simultaneously interact with the same 3D models in real-time, regardless of their physical location. This would enable team-based learning, discussions, and collaborative problem-solving, fostering a more social and interactive educational experience.

To further enhance engagement, CodeAR could incorporate gamification elements, such as quizzes, progress rewards, achievements, and interactive challenges. These features would transform learning into an enjoyable and motivating experience, encouraging students to explore more topics and improve their skills consistently.

With the rapid advancement of AR glasses, headsets, and wearable devices, CodeAR could transition from mobile and tablet-based experiences to fully immersive, hands-free learning environments. Students could manipulate virtual objects naturally with gestures, explore large-scale simulations, or even participate in virtual lab experiments without the constraints of traditional screens.

Moreover, by combining Augmented Reality (AR) with Virtual Reality (VR) and Mixed Reality (MR), CodeAR could provide hybrid learning platforms. Students could seamlessly switch between interacting with models in their real environment and exploring fully immersive virtual simulations, offering multiple perspectives on complex concepts and enhancing experiential learning.

Integration with cloud-based platforms also presents a significant opportunity for scalability. Cloud storage would enable access to a large repository of 3D assets, real-time content updates, remote learning capabilities, and cross-device synchronization. This would make CodeAR a globally accessible learning platform, capable of serving students and institutions with consistent, up-to-date educational resources.

In summary, the future scope of CodeAR lies in creating a comprehensive, adaptive, and interactive learning ecosystem that not only simplifies complex concepts but also transforms traditional education into a highly engaging, technology-driven experience. By leveraging AI, cloud computing, collaborative AR, and hybrid reality technologies, CodeAR has the potential to redefine modern education, fostering curiosity, critical thinking, and practical understanding among learners worldwide.

Future developments may include:

- AI-Integrated Learning: Implementing AI-driven content recommendations that adapt to a student's learning pace and performance.
- Voice-Interactive Learning: Adding AI-based voice assistants to explain concepts, answer questions, and guide learners during AR sessions.
- Cross-Platform Expansion: Extending compatibility to WebAR and iOS ARKit, enabling access without dedicated applications.
- Gamification & Assessments: Introducing AR-based quizzes, badges, and performance analytics to enhance motivation and engagement.
- Cloud Synchronization & Collaboration: Allowing students and instructors to collaborate in shared AR environments in real time.
- Offline Learning Mode: Integrating local caching of AR assets for uninterrupted learning without internet dependency.
- Integration with Learning Management Systems (LMS): Seamless data exchange with platforms like Moodle or Google Classroom for attendance, grading, and progress tracking.

APPENDICES

A. Source Code

Marker Recognition and Model Loading (C# Script)

```
using UnityEngine;
using Vuforia;
public classMarkerRecognition:MonoBehaviour, ITrackableEventHandler
{
    private TrackableBehaviour mTrackableBehaviour;
    public GameObject model3D;
    void Start()
    {
        mTrackableBehaviour = GetComponent<TrackableBehaviour>();
        if (mTrackableBehaviour)
            mTrackableBehaviour.RegisterTrackableEventHandler(this);
        model3D.SetActive(false);
    }
    public void OnTrackableStateChanged(
TrackableBehaviour.Status previousStatus,
        TrackableBehaviour.Status newStatus)
    {
        if (newStatus == TrackableBehaviour.Status.DETECTED ||
            newStatus == TrackableBehaviour.Status.TRACKED)
        {
            OnMarkerFound();
        }
        else
        {
            OnMarkerLost();
        }
    }
    private void OnMarkerFound()
    {
        model3D.SetActive(true);
        Debug.Log("Marker Found - Displaying 3D Model");
        FindObjectOfType<AudioNarration>().PlayAudio();
    }
```

```
    private void OnMarkerLost()
    {
        model3D.SetActive(false);
        Debug.Log("Marker Lost - Model Hidden");
}   }
```

Audio Narration Module

```
using UnityEngine;
public class AudioNarration : MonoBehaviour
{
    public AudioSource narrationClip;

    public void PlayAudio()
    {
        if (!narrationClip.isPlaying)
        {
            narrationClip.Play();
            Debug.Log("Narration Started");
        }
    }

    public void StopAudio()
    {
        if (narrationClip.isPlaying)
        {
              narrationClip.Stop();
            Debug.Log("Narration Stopped");
        }
    }
}
```

Scene Navigation Script

```
using UnityEngine;
using UnityEngine.SceneManagement;

public class SceneNavigator : MonoBehaviour
{
    public void GoToScanner()
    {
        SceneManager.LoadScene("ARScanner");
    }

    public void GoToHome()
    {
        SceneManager.LoadScene("HomeScene");
    }

    public void GoTo3DViewer()
    {
```

```
        SceneManager.LoadScene("ModelViewer");
    }

    public void ExitApp()
    {
        Application.Quit();
    }
}
```

Progress Tracking System

```
using UnityEngine;
using Firebase.Database;
using Vuforia;
[System.Serializable]
public class ProgressData
{
    public string topic;
    public bool completed;

    public ProgressData(string topic, bool completed)
    {
        this.topic = topic;
        this.completed = completed;
    }
}
public class CodeARManager : MonoBehaviour
{
    private DatabaseReference dbRef;
    public GameObject arModel;
    public AudioSource narration;
    public string topicName;
    public string userId = "User001";
    private TrackableBehaviour trackableBehaviour;

    void Start()
    {
        dbRef = FirebaseDatabase.DefaultInstance.RootReference;
        trackableBehaviour = GetComponent<TrackableBehaviour>();
        if (trackableBehaviour)
        {
trackableBehaviour.RegisterOnTrackableStatusChanged(OnTrackableStatusChanged);
        }
        if (arModel)
            arModel.SetActive(false);
    }
    void OnTrackableStatusChanged(TrackableBehaviour.StatusChangeResult statusChange)
    {
        if(statusChange.NewStatus==TrackableBehaviour.Status.TRACKD
statusChange.NewStatus==TrackableBehaviour.Status.DETECTE)
        {
```

```
            ShowModel();
            PlayNarration();
            UpdateProgress(userId, topicName, true);
        }
        else
        {
            HideModel();
            StopNarration();
        }
    }
    void ShowModel()
    {
        if (arModel)
            arModel.SetActive(true);
    }
    void HideModel()
    {
        if (arModel)
            arModel.SetActive(false);
    }
    void PlayNarration()
    {
        if (narration && !narration.isPlaying)
            narration.Play();
    }
    void StopNarration()
    {
        if (narration && narration.isPlaying)
            narration.Stop();
    }

    public void UpdateProgress(string userId, string topicName, bool isCompleted)
    {
        ProgressData data=newProgressData(topicName, isCompleted);
        string json = JsonUtility.ToJson(data);
        dbRef.Child("Users").Child(userId).Child("Progress").Child(topicName)
            .SetRawJsonValueAsync(json)
            .ContinueWith(task => {
                if (task.IsCompleted)
                    Debug.Log("Progress updated for: " + topicName);
                else
                    Debug.LogError("Failed to update progress: " + task.Exception);
            });
    }
}
```

VII. SCREEN SHOTS

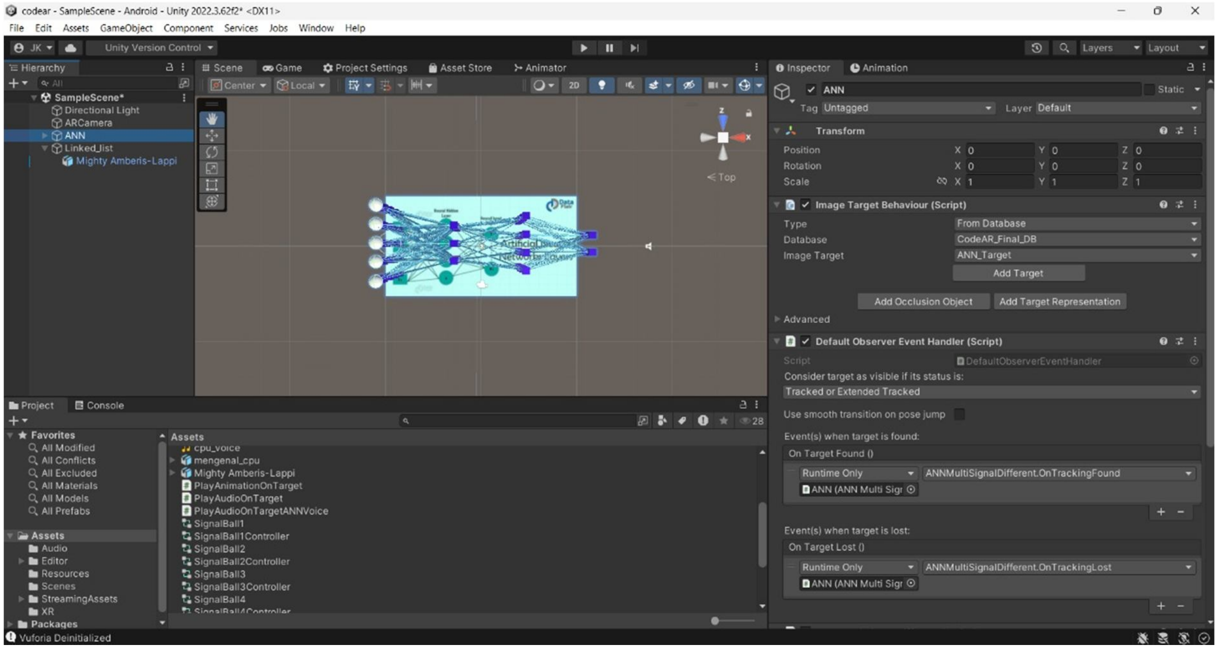


Figure 5: Home Page

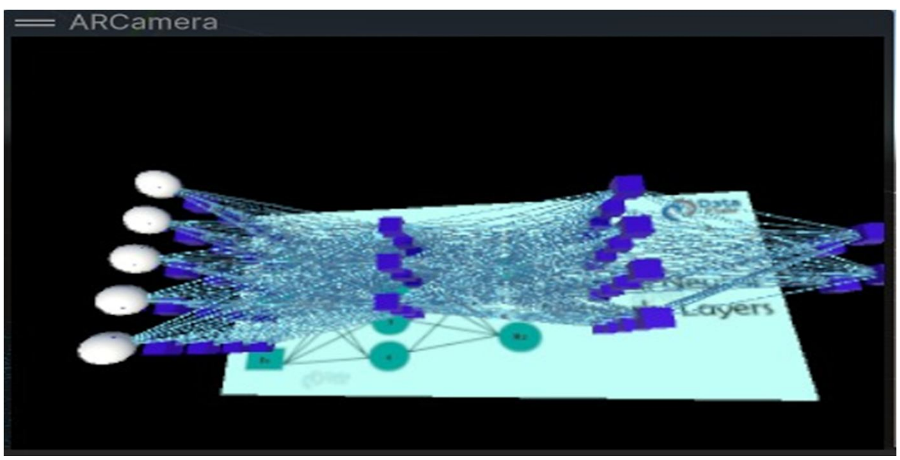


Figure 6 : AR Camara View

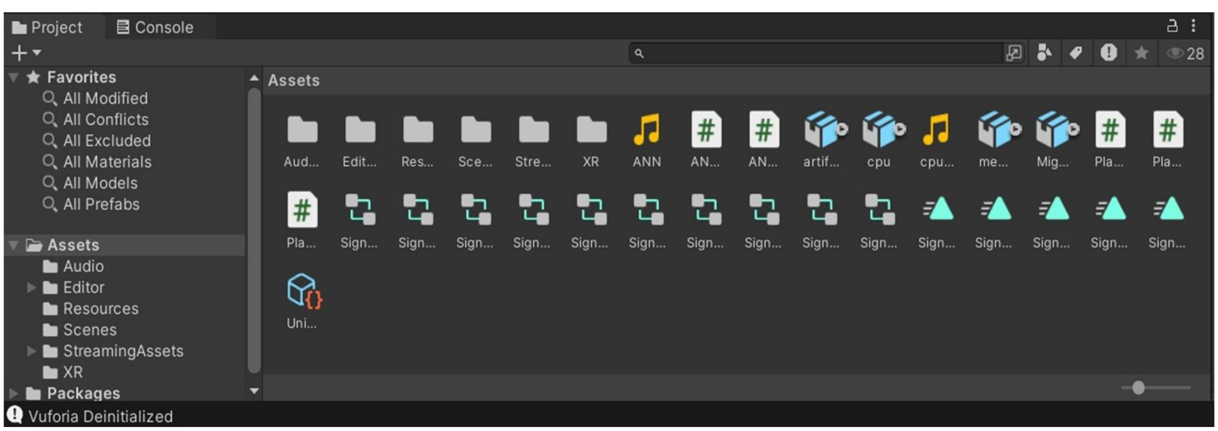


Figure 7: Assets Page

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue XI Nov 2025- Available at www.ijraset.com

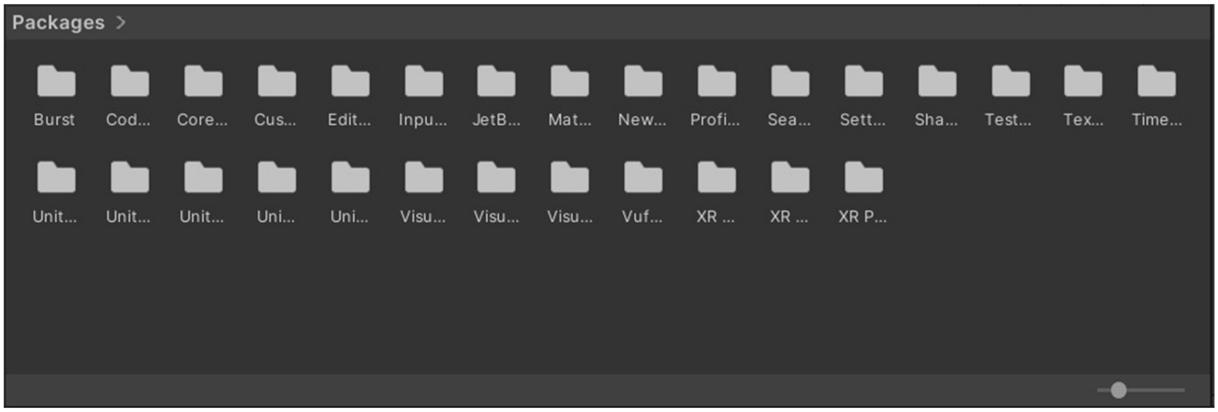


Figure 8: Package Manager

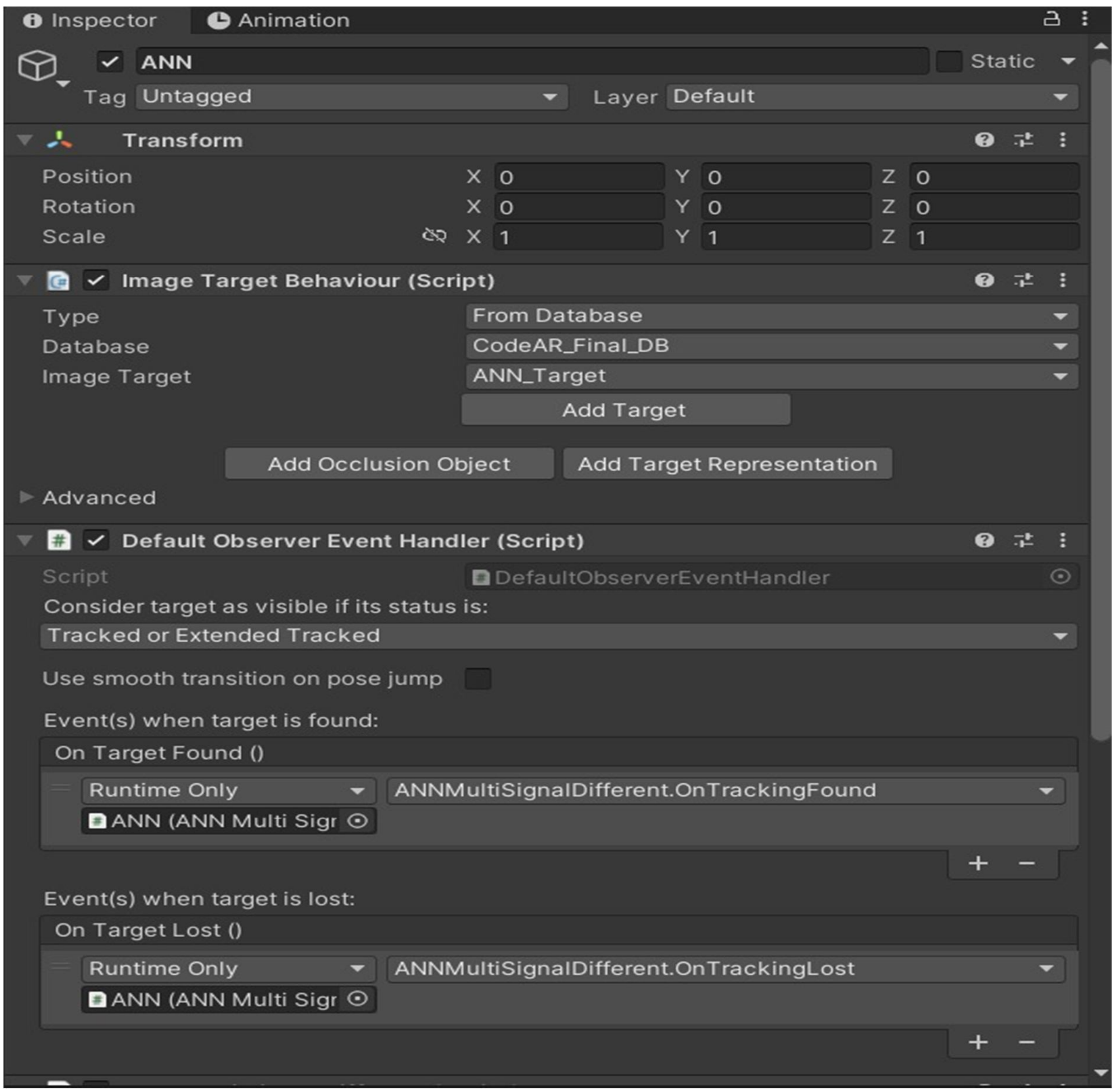


Figure 9: Package Manager

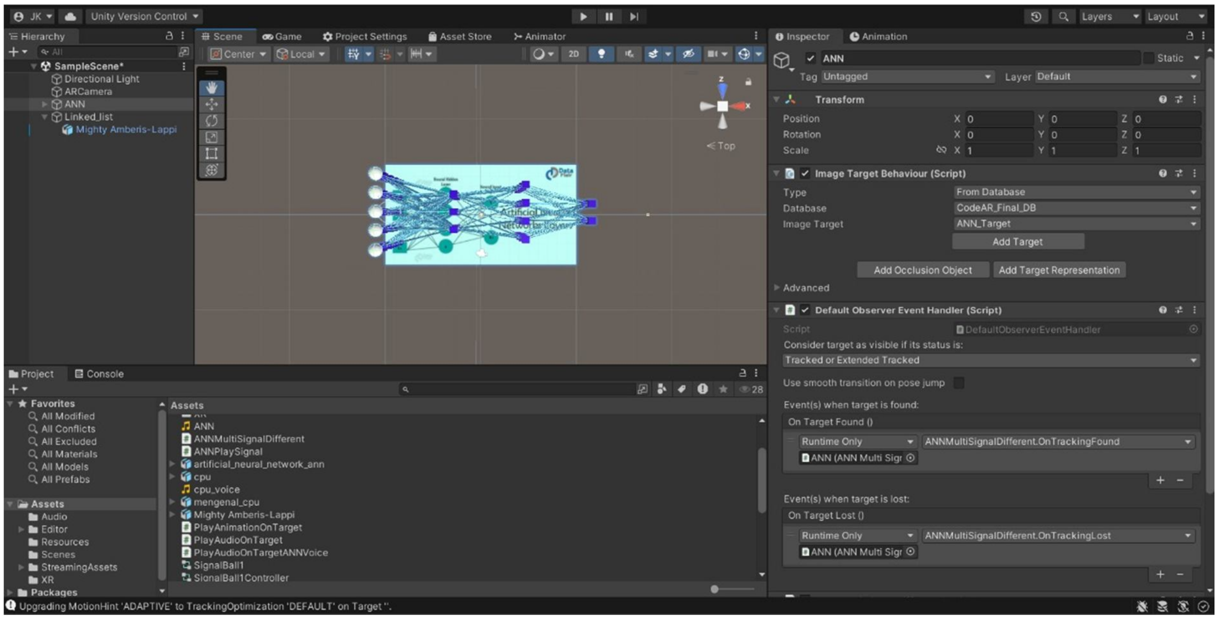


Figure 10: Testing

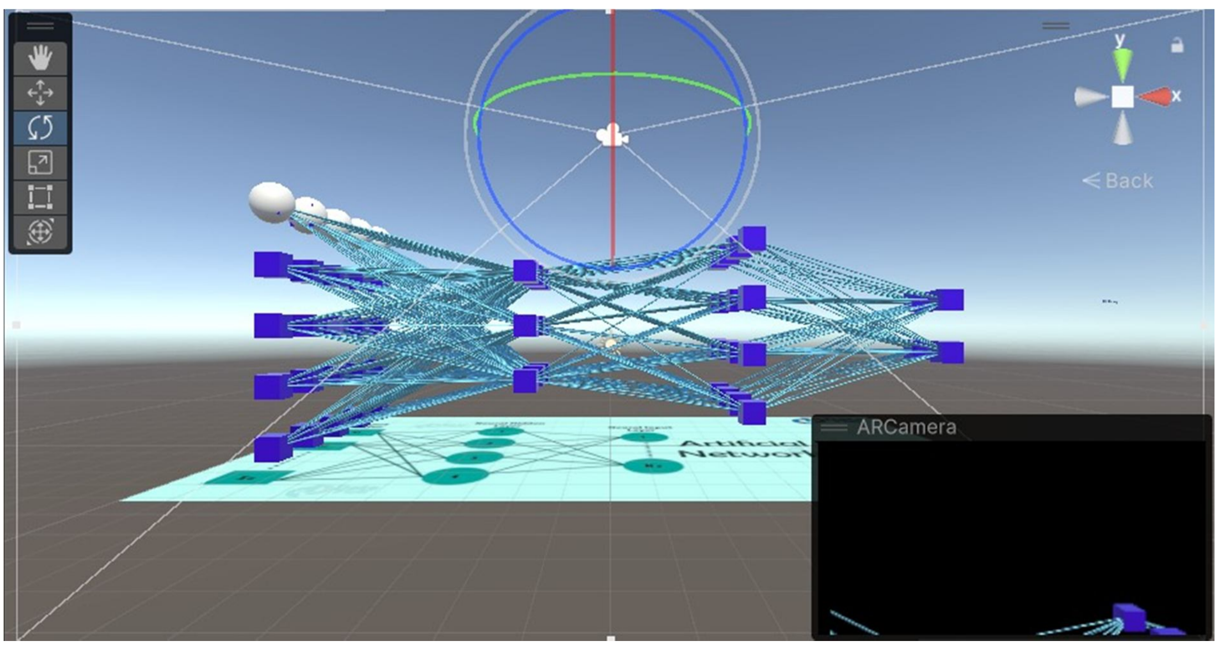


Figure 11: Final Output

VIII. ACKNOWLEDGEMENT

It is one of the most efficient tasks in life to choose the appropriate words to express one's gratitude to the beneficiaries. We are very much grateful to God who helped us all the way through the project and how molded us into what we are today.

We are grateful to our beloved Principal Dr. R. RADHAKRISHNAN, M.E., Ph.D., Adhiyamaan College of Engineering (An Autonomous Institution), Hosur for providing the opportunity to do this work in premises.

We acknowledge our heartful gratitude to Dr. G. FATHIMA, M.E., Ph.D., Professor and Head of the Department, Department of Computer Science and Engineering, Adhiyamaan College of Engineering (An Autonomous Institution), Hosur, for her guidance and valuable suggestions and encouragement throughout this project and made us to complete this project successfully.

We are highly indebted to Mrs. D.M. VIJAYA LAKSHMI, M.E., (Ph.D.,) Supervisor, Assistant Professor, Department of Computer Science and Engineering, Adhiyamaan College of Engineering (An Autonomous Institution), Hosur, whose immense support encouragement and valuable guidance were responsible to complete the project successfully.

We also extent our thanks to Project Coordinator and all Staff Members for their support in complete this project successfully.

Finally, we would like to thank to our parents, without their motivational and support would not have been possible for us to complete this project successfully.

REFERENCES

[1] Billinghurst, M., Clark, A., & Lee, G. (2015). A Survey of Augmented Reality. Foundations and Trends® in Human–Computer Interaction.

[2] Craig, A. B. (2013). Understanding Augmented Reality: Concepts and Applications. Morgan Kaufmann Publishers.

[3] Schmalstieg, D., & Hollerer, T. (2016). Augmented Reality: Principles and Practice. Addison-Wesley Professional.

[4] Milgram, P., & Kishino, F. (1994). A Taxonomy of Mixed Reality Visual Displays. IEICE Transactions on Information and Systems, E77-D(12), 1321–1329.

[5] Unity Learn. Developing Augmented Reality Apps with Unity and Vuforia. Retrieved from https://learn.unity.com

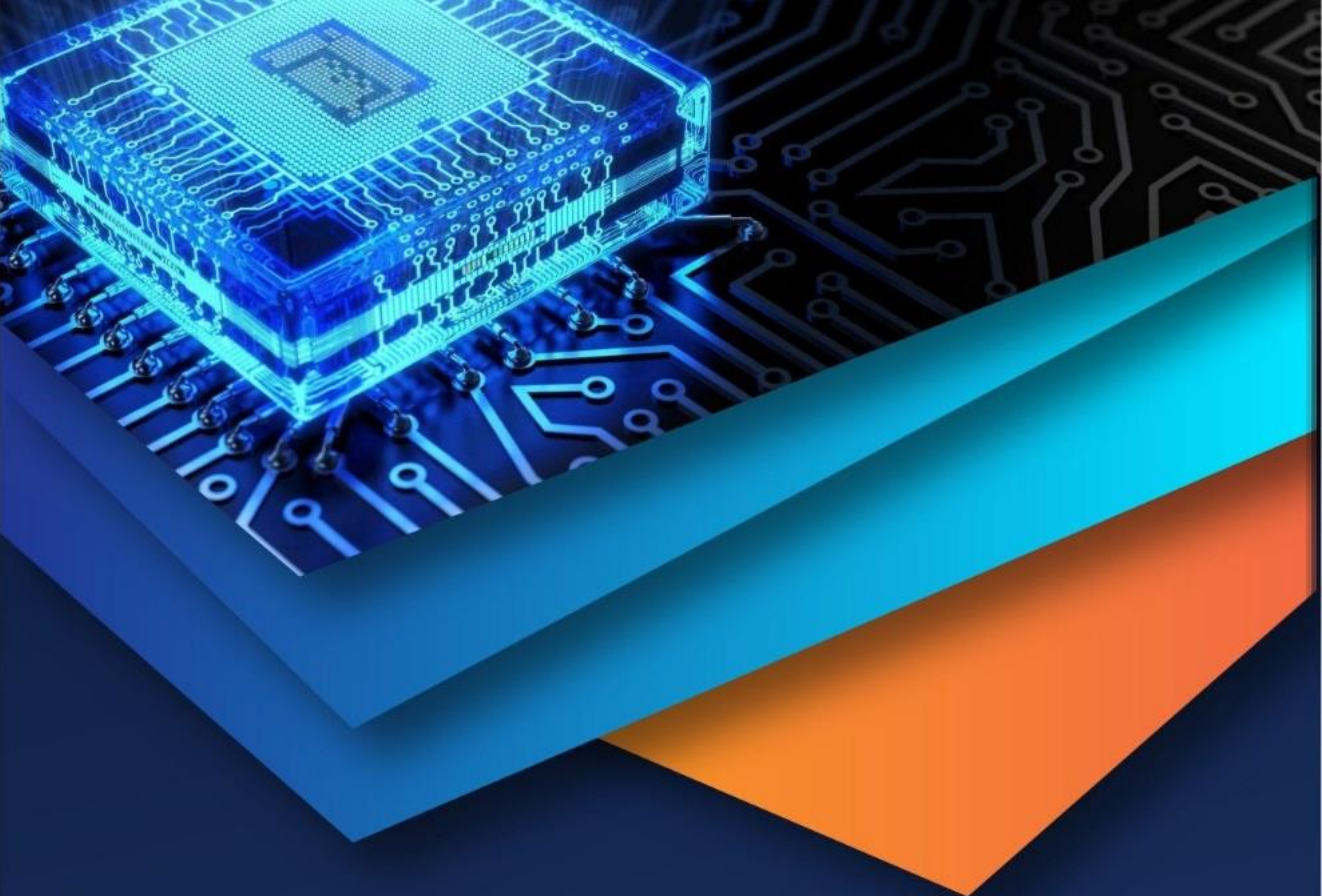
[6] Zhou, F., Duh, H. B. L., & Billinghurst, M. (2008). Trends in Augmented Reality Tracking, Interaction and Display: A Review of Ten Years of ISMAR. Proceedings of ISMAR 2008.

[7] IEEE Xplore Digital Library. AR in Education and Learning Technologies. Retrieved from https://ieeexplore.ieee.org

[8] ResearchGate. Applications of Augmented Reality in Engineering Education. Retrieved from https://www.researchgate.net

[9] Make Cloude Team. CodeAR – AR-Based Learning System Documentation. Internal Project Report, 2025.

[10] Chen, C.-M., & Tsai, Y.-N. (2020). Interactive augmented reality system for enhancing learning motivation and performance in science education. Educational Technology & Society, 23(1), 45–58.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY