WKU XR

S.T.E.A.M. Resource Guide

For Educators & Students

Version 1.0

TABLE OF CONTENTS

1 **Introduction** 3
Getting Started 3

2 **3D Modeling Resources** 5
Blender 6
Shapr3D 7
Spline 8

3 **Programming Resources** 9
Scratch 10
Mimo App 11
Python 12

4 **Game Development Resources** 13
Unity 14
Gamefroot 15
GDevelop 16

6

Augmented & Virtual Reality Resources 17

Adobe Aero App 18

Google Cardboard 19

YouTube VR 20

7

Conclusion 21

Finding More Resources 21

Contact 22

Acknowledgements 22

Terminology Guide 23

INTRODUCTION

This resource guide serves as a starting point for educators and students interested in developing skills utilized in the WKU Extended Reality (XR) Research Lab. The software and hardware described in this guide allow interdisciplinary innovators (you!) to build on **3D modeling, programming, video game development, and virtual/augmented reality technologies** - regardless of age, education, or skill.

Given our interdisciplinary approach to research, WKU XR understands the value of STEAM education. We truly believe that continued collaboration between STEM and the Arts will drive the future of technology. However, we also understand that in order to have this reality, the concept of **STEAM** research and education needs to become more commonplace in schools, universities, and the workplace. This is where you come in! Learning new skills that can be combined with your existing capabilities makes you a more holistic individual and makes you an interdisciplinary innovator in the world of STEAM research.

GETTING STARTED

Each resource is accompanied by a brief summary, system requirements and usage guidelines, pros & cons, and a recommended beginning tutorial. This guide has been organized in a way that mimics the real-life process of VR development from 3D modeling and programming to game development and VR integration. Additionally, a difficulty level is associated with each resource so that the researcher may learn at their own pace.



Finally, words that have been colored **red** indicate that a more concise definition is found at the end of the guide in the **terminology guide**.

3D MODELING RESOURCES

One of the most fundamental skills required for extended reality development is having an understanding of the 3D environment and the functionality of 3D models. **3D modeling** itself is a unique medium because it is one of the main touchpoints between STEM & the Arts in industry. Aside from having the skills necessary to digitally "sculpt" objects, 3D artists often also require a sophisticated understanding of the system requirements of the software they are using. Like programming, 3D modeling requires patience, trial-and-error, and critical thinking, however, this does not mean that it has to be difficult to learn.



3D Model of a Moai Head created by Madison Whittle. Was later featured in the Lab's Mini Golf Game.

In the WKU XR Lab, we have students from STEM and Art backgrounds with experience in 3D modeling. WKU's Computer Animation and Game Design programs allow students of all disciplines to learn 3D modeling for animation and video game applications, respectfully. The software standard taught at WKU is Maya, the same program used to create the fictionalized characters in James Kameron's *Avatar*. In the lab, however, students have the option to choose the program that they are most comfortable with and that best fits their needs.

BLENDER 

Blender is a free 3D modeling software that remains one of the most popular programs of its kind. Although WKU's Art Department currently teaches Maya (a more expensive, less user-friendly software), being able to learn the functionality and usability of 3D models will give researchers a leg-up in 3D modeling/animation/CAD.

SYSTEM REQUIREMENTS

(*Minimum*)

Operating system: Windows 8.1+, MacOS 10.13+, or Linux

Storage space: ~500 MB

Display: 1280×720, 24 bit

CPU: 64-bit dual-core with SSE2 support, 2 GHz

RAM: 8 GB

Graphics card: Any GPU card operating on OpenGL 3.3 GPU with 2 GB RAM

Mouse: Standard mouse, trackpad, or pen and tablet

PROS & CONS

Pros:

- Free and open-source software, tutorials and assets available online
- Sophisticated program; Allows users to artistically manipulate models and have free control over sculpting

Cons:

- Advanced user interface; may be intimidating to first-time users
- Decently steep learning curve

BEGINNING TUTORIAL

The recommended beginning tutorial for Blender is the Blender Guru's Donut series on YouTube: https://www.youtube.com/watch?v=nIoXOplUvAw

SHAPR3D  

Shapr3D is a 3D design/CAD program that works on iPad, macOS, and PC. It is used mainly for engineering applications, however, it can also be used as a more rigid 3D design software (as opposed to Blender). Additionally, according to Shapr3D's website, a subscription is free for students and educators.

SYSTEM REQUIREMENTS

See the link for more information on compatible devices:

https://www.shapr3d.com/compatible-devices

PROS & CONS

Pros:

- Free subscription for students and educators
- Sophisticated program; Allows users to artistically manipulate models and have free control over sculpting

Cons:

- More rigid functionality; Limited creative freedom compared to Blender
- Minimal options and controls; Not as advanced

BEGINNING TUTORIAL

The recommended beginning tutorial for Shapr3D is: https://productdesignonline.com/learn-shapr3d-in-10-days-for-beginners/

SPLINE 

Spline is a free, browser-based 3D modeling software and app that allows users to work on 3D modeling from any computer. Its features include basic modeling capabilities, easy texturing and lighting, animation, and collaboration between parties. Compared to Blender, the **user interface** is more condensed and more visually friendly.

SYSTEM REQUIREMENTS

(Available for use on the following platforms)

Download software here: https://spline.design/

Browser

macOS (m1)

macOS (intel)

Windows

Linux

PROS & CONS

Pros:

- Free and open-source software, tutorials and assets available online
- Sophisticated program; Allows users to artistically manipulate models and have free control over sculpting

Cons:

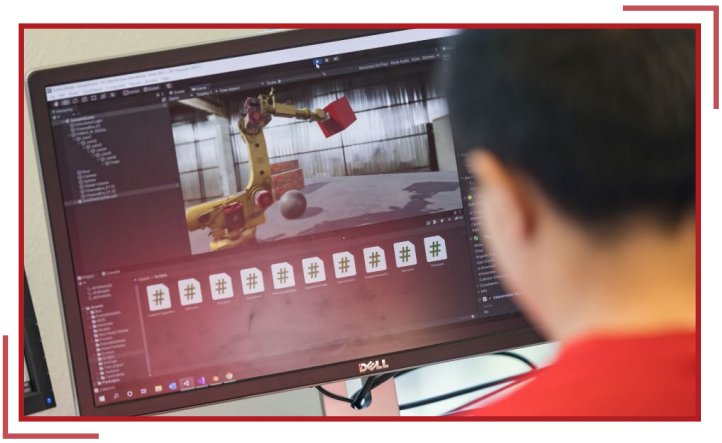
- Advanced user interface; may be intimidating to first-time users
- Decently steep learning curve

BEGINNING TUTORIAL

The recommended beginning tutorial for Spline is the School of Motion's introduction on YouTube: https://www.youtube.com/watch?v=rfqxhriBodc

PROGRAMMING RESOURCES

Of the skills detailed on this list, programming may be the most daunting to beginner interdisciplinary innovators. Simply put, **programming** is computational problem solving using a particular programming language. Similar to how human language is a conversation between two individuals, **programming languages** enable humans "talk" to computers. And, much like learning a second language, there are vital nuances one must recognize to be an effective "translator;" Whereas some human languages utilize gendered nouns and specific sentence structure, programming languages are highly particular about symbols, defining and organizing variables, and case-sensitivity.



Researcher Diego Minaya looks over his programming for the KY NSF EPSCoR Grant.

An obvious benefit of an interdisciplinary space is the availability of several skills. This being said, computer science students bring the knowledge and understanding of a variety of programming languages that assist in VR development. However, while the majority of CS students have coding experience in other languages, several of them have found themselves learning a new language (C#) to use our preferred game development software. For the purpose of this research guide, we see that the general understanding of programming is more valuable than diving headfirst into one specific language and recommend the provided resources for beginner to intermediate programmers. For more advanced programmers interested in game development, we recommend brushing up on C#.

SCRATCH  

Scratch is arguably the easiest, most user-friendly introduction to programming there is. Developed by MIT, Scratch uses **block coding** that allows users to drag-and-drop code like puzzle pieces to visually "build" a program. Though developed with 8-16 year olds in mind, Scratch is an easy and delightful way for beginners of all ages to jump into the world of programming.

SYSTEM REQUIREMENTS

Desktop

Chrome (63+)

Edge (15+)

Firefox (57+)

Safari (11+)

Internet Explorer is NOT supported.

Tablet

Mobile Chrome (63+)

PROS & CONS

Pros:

- Free, browser-based application with a library of assets and online community
- Visual-based editor; No typed code required

Cons:

- Though it does teach the foundation of coding, it does not introduce **text-based programming** languages
- Limited functionality and features

BEGINNING TUTORIAL

The recommended beginning tutorial for Scratch is the Scratch Tutorials on their website: https://sip.scratch.mit.edu/hoc-2019/

MIMO APP





The Mimo App for Coding/Programming turns a daunting skill into a palatable, enjoyable activity. Utilizing **gamification**, Mimo assists users in learning programming by encouraging five-minute exercises every day in the area of their choice (Web development, Python, and SQL). Much like the language-learning app Duolingo, Mimo includes fill-in-the-blanks, multiple-choice, and challenge exercises.

SYSTEM REQUIREMENTS

Android

6.0 and up

iPhone

iOS 14.0 or later

PROS & CONS

Pros:

- Ideal for beginners to understand functionality of text-based languages
- Gamification makes learning enjoyable and allows for quick, trackable learning

Cons:

- Requires in-app purchases to unlock full functionality
- Does not replace the experience gain by practicing actual programming

BEGINNING TUTORIAL

The Mimo App is avaliable on both the App Store and Google Play Store.

Google Play Store: https://play.google.com/store/search?q=mimo&c=apps&hl=en_US&gl=US

App Store: https://apps.apple.com/us/app/mimo-learn-coding-programming/id1133960732

PYTHON





Python is one of the most comprehensible, easy-to-learn text-based programming languages to date. It offers the hands-on experience of writing code and understanding the programming process without overly complicated **syntax**. There are over 200,000 Python packages (add-ons) that you can download to increase the possibilities of the program.

SYSTEM REQUIREMENTS

Operating system: Linux- Ubuntu 16.04 to 17.10, or Windows 7 to 10, with 2GB RAM
(4GB preferable)

PROS & CONS

Pros:

- Ideal for beginners to understand functionality of text-based languages
- Free, open source software with 200,000 add-ons and thousands of online resources

Cons:

- Python does not offer integrated tutorials, so it is recommended for new users to find tutorials/prompts
- While great for learning, the functionality of Python may not be as sophisticated as other languages

BEGINNING TUTORIAL

A quick beginners tutorial can be found on W3 schools: https://www.youtube.com/watch?v=nIoXOplUvAw

GAME DEVELOPMENT RESOURCES

Video Game Development is where 3D modeling and programming combine so that the final product may be put into extended reality. **Game engines** (or game development software) allow game developers to place their custom or **prefab** (pre-fabricated) models into the workspace so that gamification can take place via programming. While a large part of gamification is the actual development of a video game, it also has to do with the psychology between the player and their interaction with the game. Gamification has become increasingly popular over the past few decades; As you may have noticed, there has been a large initiative to "gamify" education, especially within schools and the military.



Researcher Chris Nalani DiMeo demonstrates a hand tracking program they developed in Unity for the KY NSF EPSCoR Grant.

At the XR Lab, we implement both game development and gamification to bring our projects to life. In the preliminary stages of a project, our XR researchers utilize their knowledge of user experience and gamification to create an innovative and enjoyable end product. This research includes but isn't limited to, primary and secondary research, interviews, referencing other examples of gamification, application, and user testing. After this foundation is established, 3D models are created and brought into Unity, the lab's game engine of choice, where programming and output to virtual reality take place.

UNITY  

Unity is one of the most popular game engines in the industry with nearly 50% of all mobile games utilizing the software. Since its release in 2005, Unity has served as a cross-platform 2D and 3D game engine capable of creating for 25 different gaming platforms. As of 2022, about 750,000 games have been developed using Unity, including Beat Saber, Pokemon Go, and Among Us.

SYSTEM REQUIREMENTS

(For recommended LTS (long term support) version 2019.4.17)

Please visit this link for specific system requirments: https://unity3d.com/unity/whats-new/2019.4.17

PROS & CONS

Pros:

- Free, open source software with a library with thousands of prefabs
- Visual-based editor; Depending on the project, little to no code is required

Cons:

- Currently, all games you create must be from scratch; no template support
- If you are developing a more advanced project, an understanding of the coding language **C#** is required

BEGINNING TUTORIAL

Unity offers a section on their program called Unity Learn that gives users step-by-step instructions on how to build games. For example, this is the “Creating With LEGO Tools” tutorial to get started: https://learn.unity.com/tutorial/creating-with-lego-tools#

GAMEFROOT





Gamefroot is an online cloud-based platform for making 2D video games. As proponents of accessibility, Gramefroot states on their website, “We combine a powerful visual coding interface and an easy-to-use level editor to take game development to the masses in a culturally rich and engaging environment.” In fact, Gamefroot utilizes the same coding blocks as Scratch!

SYSTEM REQUIREMENTS

Operating System

Windows 8 / 8.1 / 10

macOS 10.11 - 11

RAM

4 GB or higher

Hard Drive

5 GB or more of free storage

PROS & CONS

Pros:

- Incredibly user-friendly; Intended for first-time game developers
- Prescence of online communtiy; Ease access to games made by other developers in the arcade

Cons:

- Great for learning the functionality of game design but limits more advanced users
- Requires a paid subscription to unlock certain features

BEGINNING TUTORIAL

Gamefroot offers a collection of beginning tutorials and resources on their website: https://make.gamefroot.com/dashboard/home

GDEVELOP





GDevelop is ideal for the individual that is too advanced for Scratch but is not quite ready for Unity. Unlike Gamefroot, GDevelop offers more refined customization of assets via integrated design functions and JavaScript support. Additionally, there is a wide variety of resources available both on the main menu as well as inside of menus of your project, allowing for easy skill development and problem-solving.

SYSTEM REQUIREMENTS

CPU:

any 64bit compatible CPU

RAM:

1GB RAM (for GD itself, the game you are making depends on you)

GPU:

any WebGL compatible graphics

OS:

any 64bit operating system, the newer the better

PROS & CONS

Pros:

- Powerful game engine given its simplicity; Access to variety of features
- Can be as simple or complex as the user would like thanks to JavaScript editing

Cons:

- Some users believe that the multitude of features is overwhelming
- Certain features are block by a subscription paywall

BEGINNING TUTORIAL

In addition to the integrated tutorials in GDevelop's program, freeCodeCamp.org offers a crash course on ther=ir YouTube Channel: https://www.youtube.com/watch?v=iHF5fwsqu4I

VIRTUAL & AUGMENTED REALITY RESOURCES

A quickly emerging feat in the tech world today is extended reality technologies. **Extended Reality (XR)** is an umbrella term that describes technology that combines reality with "fabricated" reality, which includes virtual and augmented realities. **Virtual Reality (VR)** utilizes a headset to "fully immerse" the user in a world that is not their own, however, head and body movements, walking IRL, and even eye and finger movements register in VR, giving the illusion of immersion. Examples of VR include BeatSaber and Facebook's Oculus Headset. **Augmented Reality (AR)**, on the other hand, utilizes a smart device to overlay visuals into the real world. Examples of AR include Pokemon GO and Snapchat filters.



The Mini Golf Game was the first collaborative project executed in the lab. Researchers Chris Nalani DiMeo, Diego Minaya, Zachary Mers, Sarah Terry, Madison Whittle, Apirada Chetawatee, and Isabel Ocegueda contributed their talents to this project.

As an Extended Reality lab, we work in several varieties of XR technology, including VR, AR, and MR (Mixed Reality). The most popular medium we use is VR, which we develop using the Oculus Quest 2 headset due to its development capabilities and cost-effectiveness. If you are someone who is interested in pursuing game development, we recommend purchasing your own Quest 2 headset. However, if this is not a practical option for you or you are unsure if this skill interests you, we have free alternatives that will help kickstart your journey into VR/AR!

ADOBE AERO APP





Currently, the Adobe Aero App is free software because it is in Beta. It allows users to upload images/video/3D files into the app, add time-based animations, and then view their creations in augmented reality using specialized QR codes. Ideally, artists could use a drawing program such as Adobe Illustrator or Procreate to draw something with layers and then export each individual layer as a .png.

SYSTEM REQUIREMENTS

Desktop Requirements: https://helpx.adobe.com/aero/system-requirements-macos-windows.html

Smart Phone Requirements: https://helpx.adobe.com/aero/system-requirements.html

PROS & CONS

Pros:

- User-friendly; No coding involved
- Animating and sharing content is made incredibly accessible

Cons:

- Because it is an Adobe product, there is a chance it will not be free in the future

BEGINNING TUTORIAL

The recommended beginning tutorial for Adobe Aero is Adobe's beginning page on their website: https://helpx.adobe.com/support/adobe-aero.html

GOOGLE CARDBOARD  

If purchasing a Quest 2 Headset is not an option, Google Cardboard is a cheaper alternative that mimics most of the functionality of a real VR headset. The Google Cardboard is a device in which the user can place their smartphone in the "headset" to see 360° games, pictures, and videos. Though Google has stopped selling the Cardboard Headset itself (the app is still available as of June 2022), there are several alternative headsets available for purchase here: https://arvr.google.com/

SYSTEM REQUIREMENTS

Getting Started with the Headset & App: https://www.businessinsider.com/google-cardboard

PROS & CONS

Pros:

- Inexpensive ($10-$15 a headset)
- Is able to emulate VR with most smartphones

Cons:

- Quality of VR experience is dependent on phone
- Lacks the functionality of a high-end VR headset

BEGINNING TUTORIAL

The Google Cardboard Quickstart Guide provided by UC Santa Cruz's University Library shows how to set up the headset and recommended apps: https://guides.library.ucsc.edu/DS/Resources/GoogleCardboard

YOUTUBE VR  

YouTube VR is a genre of videos on YouTube in which viewers can watch 360° videos both on desktop and compatible VR devices (such as the Oculus or Google Cardboard headsets).

USAGE INFORMATION

- 360° videos can be viewed on normal devices such as a smartphone and/or desktop. You can use your cursor or fingertip (depending on the device) to move the screen as the video plays.
- Additionally, if you have a VR headset, you can either access YouTube VR on a higher-end headset (i.e. Oculus Quest 2) or pull up a 360° video on your YouTube smartphone app, turn the video horizontally, and press the small headset icon to enable headset mode. This will bring make the video split-screen to give the illusion of virtual reality once placed in the headset of choice. Here are some more examples of low-end headsets: https://www.aniwaa.com/buyers-guide/vr-ar/best-smartphone-vr-headset/

PROS & CONS

Pros:

- Free resource
- Thousands of VR videos

Cons:

- Quality of VR experience is dependent on phone
- Requires a headset (high or low end) to get the “true” VR experience

BEGINNING TUTORIAL

Begin exploring virtual reality videos on YouTube: https://www.youtube.com/channel/UCzuqhhs6NWbgTzMuM09WKDQ

CONCLUSION

We are in a unique position now where STEAM education is rapidly growing, which means that we are actively contributing to an emerging field. Whether you are a hobbyist that is interested in game development, a student wanting to learn how to program, or an educator introducing your class to STEAM concepts, you are contributing to the future of interdisciplinary innovation!

In closing, there is one more suggestion we would like to share in this guide. Whatever you create using the skills you've learned, share it! Whether you post your creation on social media or share it with your peers, you are publishing your work to the world and showing the importance of STEAM and XR research.

FINDING MORE RESOURCES

This guide is intended to get you started down the pathway of STEAM & XR education, not limit it! Because everyone learns differently, we encourage you to find the guides and resources that work best for you - whether that is a video, article, or interactive tutorial. Additionally, online communities exist for nearly every resource detailed in this guide.

As always, the WKU XR Lab is dedicated to the education and application of XR technologies locally and beyond. If you are an educator or student in the area, please reach out to schedule a visit to our lab or have a group of researchers come to your school. For more specific questions, please reference the contact information found on the following page.

CONTACT

Prof. Kristina Arnold

Art & Design Department Head

kristina.arnold@wku.edu

(270) 745-6566

Dr. Greg Arbuckle

Mechanical Engineering Technology

greg.arbuckle@wku.edu

(270) 745-2403

Prof. Mark Simpson

User Experience

mark.simpson@wku.edu

(270) 745-5838

Dr. Michael Galloway

Computer Science

jeffrey.galloway@wku.edu

(270) 745-2859

Prof. Truth Tran

Computer Animation

truth.tran@wku.edu

ACKNOWLEDGEMENTS

WKU PCAL

WKU DEPARTMENT OF ART & DESIGN

WKU Ogden Science & Engineering

WKU SCHOOL OF ENGINEERING & APPLIED SCIENCES

WKU Office of Research & Creative Activity

WKU Office of the President

TERMINOLOGY

3D Modeling	The use of computer graphics to create replications of objects digitally.
Augmented Reality	Also known as AR; A form of Extended Reality in which a device is used to overlay computer graphics onto the real world (For example, Pokemon Go).
Block Coding	A type of programming that uses pre-made, puzzle-piece like blocks of code to create a program (ex. Scratch).
C#	(Pronounced 'C-Sharp'); The programming language used to develop in the game engine Unity.
Extended Reality	Also known as XR; An umbrella term that describes all forms of immersive technology including virtual and augmented realities.
Game Engine	The software in which video games are developed.
Gamification	The integration of elements found in games (badges, points, levels, etc.) into systems that are not often associated with being enjoyable, such as learning a language with Duolingo.
Prefab	"Pre-fabricated" objects that can be placed and reused inside game engines during video game development.
Programming	The process of writing code to manipulate the functionality of a piece of technology.
Programming Languages	The variety of code in which you are using to write a program. Examples include C#, JavaScript, and Python.

STEAM	STEM + Arts = STEAM; The combination of Science, Technology, Engineering, Arts, and Mathematics.
Syntax	The set of rules for a given programming language; Dictates whether the combination of symbols and structure will be processed correctly.
Text-Based Programming	Unlike block coding, text-based coding requires the programmer to manually type code to write a program.
User Interface	Also known as UI; The visual part of a website, app, or game that is what the user sees and interacts with.
Video Game Development	The process of creating a video game; This includes applying 3D modeling and programming within a game engine.
Virtual Reality	Also known as VR; A form of Extended Reality in which a headset is used to fully immerse the user in an environment that is not their own.



2022

Written and Designed by Madison Whittle