Product Requirements Document
EasyMath

Matthew Chun (mwc22@cs.washington.edu),
Jin An San (jinansan@uw.edu),
Esteban Posada (epm105@uw.edu),
Tanuj Jhawar (tjhawar@uw.edu)
4/16/19

Summary

EasyMath is an AR 3D graphing assistant that uses the Magic Leap to assist students in better understanding how equations for common shapes in three dimensions are constructed.

Project Description

Multiple researchers in the mathematics community have pointed out across the world, teaching 3D geometry to high school students has been a struggle in the education system for decades. As math gets more abstract and harder to visualize it is difficult for students to grasp the new concepts and perform well. This is a real problem and one that schools around the world still face today. This is evident by the data. According to the The Third International Mathematics and Science Study (TIMSS) study from 41 nations and our own National Assessment of Educational Progress (NAEP) data, geometry is an area of dismal performance for students at all levels. Student's struggle with this type of math is why we want to tackle this problem. Our team, which is composed of three members that have been teaching assistants at the University of Washington immediately felt the huge passion and motivation to use the new AR skill sets to develop a solution that could help students learn better 3D geometry. The team also wants to provide future teachers a tool to teach better. We truly believe that education is key for the development of our world and being able to provide a product that will make an education area better taught will setup the next generation better for the future.

Therefore, we introduce EasyMath, a world-class 3D geometry AR assistant made by students for students. EasyMath will be the best way for high school students to learn everything about the 3D space. Students struggle to visualize shapes in three dimensions, so EasyMath will offer a graphing feature to plot shapes from equations. Although this is a core feature, this is not what makes us best in world. Taking advantage of the features in AR, the student will be able to interact with the shape and see how the equation changes. EasyMath is able to convert from shapes back to equations which gives students a deeper understanding of the equations. A common question in math classes is how shapes intersect with each other. EasyMath enables students to plot a number of shapes at the same time. This allows students to see how the different shapes intersect with each other. The different shapes will be created through a web interface where students can input equations to display on the Magic Leap and also see equations update in real-time when the student resizes or moves shapes using the Magic Leap controller.

The team will create EasyMath by first using the team's collective experiences in data visualization, teaching, frontend and backend development. Also, EasyMath will take advantage of the state of the art Magic Leap device to produce the AR components. The team will focus on supporting a specific, common set of shapes in 3D. For each of the shapes, the team will create the functionality to convert from equation to shape and shape to equation. Also, the team will provide the capabilities to resize and move shapes. We will first look into converting equations to shapes using mesh rendering. For the case of shape to equation, we will restrict the ways the shape can be resized to common axes and using this information can update the equation accordingly. Lastly, to create EasyMath, the team will need to create the web interface to input equations and see equations update. There are number of ways to setup the connection between the web interface and the Magic Leap. As latency could be an issue, we will need to experiment with a number of techniques to determine the optimal technique.

User Experience

Francisco is a middle school student who is working on his math assignment. He is a visual learner and he does not understand how certain equations relate to certain shapes. Specifically, he is working on a problem that asks him to explain the properties of an ellipsoid as well as similarities and differences between an ellipsoid and a sphere. However, he is having a lot of trouble visualizing this. We envision Francisco to go through the following steps to solve the problem and to discover more on his own:

1. He puts on his Magic Leap and opens EasyMath.
2. On his computer or phone, he goes to the EasyMath web interface, and enters in an equation for an ellipsoid.
3. In front of him, an ellipsoid appears, which he can interact with. This includes stretching and contracting the ellipsoid. By making changes to the ellipsoid, he sees the shape and equation change in real-time, from this, he can see how the equation for an ellipsoid relates to its properties.
4. Now, to see the similarities between an ellipsoid and a sphere, he enters in the equation for a sphere, and now he sees in front of him both an ellipsoid and a sphere.
5. He stretches and contracts the ellipsoid in XYZ directions until the length in all directions are equal. He sees that that the equation becomes similar to that of the sphere, but with a

different numbers. Now, he reshapes the sphere so that the lengths are equal. Finally, he sees that the equations are now exactly the same.

6. Out of curiosity, he wonders what it would look like to have these two shapes intersect. So, he intersects the shapes, zooms in on the intersection, observes the coordinate values, and is absolutely amazed to see that the intersection is actually a circle.
7. Even more curious now, he slides the shapes around, sees different types of intersections, and how the size of the intersection changes.
8. Super excited, he tells his friend, who is also in his math class, to put on his Magic Leap, and shows him his findings.
9. However, they still don't really understand how the volume of an ellipsoid is calculated. So, they start a group call with their other classmate, and she explains in a global shared view to them how the volume of an ellipsoid is calculated by showing them the proof with drawings and equations.
10. After they end the group call (stretch goal), Francisco is still intrigued by these shapes, so he continues to play with EasyMath, and builds real life objects (with gravity) using ellipsoids, spheres, cylinders, and prisms with the features EasyMath offers.

Hardware Platform/Device

First Option: Magic Leap because it allows us to plot in the 3D world of the user. The real 3D experience will be possible using the Magic Leap.

Backup Option: ARCore because it is more accessible for the critical mass. We will still face the problem that we wanted to address: explain 3D math in a 2D world, that is why we believe Magic Leap is our way to go and were we see the future of this product.

Deliverables

MVP

Using a web interface, students can enter in simple equations in 3D and see the shape in the Magic Leap. In the MVP we will initially start by supporting spheres and cylinders. EasyMath will support students adding multiple shapes at the same time to see how they interact.

With all the shapes we support, EasyMath has the following two features:

1. Input any variation of the shape as an equation and see it plotted on the Magic Leap.
2. Interactive Features - stretch and contract the shape and see the equation change in real-time.

We estimate to finish the first feature for the two shapes in a week and afterwards the second feature (interactive features) in a week.

Target Product

In the MVP, we only allowed the student to enter in the equation for a sphere or cylinder. Now, to complete our product, we allow the student to enter in equations for more complex shapes that include:

1. Cone
2. Ellipsoid

We also want to add must-have features like resizing the shapes, being able to use the Magic Leap controller to look at coordinate values, and even show the students intercepts for multiple graphs.

We estimate to finish this portion in 3 weeks.

Stretch Goals

We have two stretch goals:

1. **Video chat feature** - the student could call an explainer, a student or teacher. Using the Magic Leap's camera, the explainer will be able to see what the student is looking at. The explainer will be video chatting over an iPad. The explainer can draw on the screen and the annotations will show up on the field of view of the student. The challenge here is when the student looks away. To deal with this case, when we detect the student turns their head beyond a certain threshold, we will just delete the annotations made by the explainer. The next level of stretch goal would be to keep the annotations there when the student looks away and looks back at the original location.
2. **Multi-student mode** - a great use case location for EasyMath is in the classroom. In the classroom there are a number of other students and learning in groups is a great way to speed up a student's learning. Therefore, a stretch goal is to create a multi-student mode where all students see the same "world". Students can add shapes to the world by inputting a mathematical equation and see how their shapes interact with shapes other students created. This mode is superior in AR than VR because teachers could assign problems or task for groups of students to do in class. AR enables the students to see the teacher and the instructions, that will usually be on paper, a tablet, or computer, at the same time as they see the shapes.

Performance Metrics

We will evaluate our product on the following factors:

1. **Functionality** - success will be defined as completing all the features specified in the Target Product. Exceeding our expectations would be accomplishing any of the stretch goals.
2. **Performance and Accuracy** - we define success under this metric as being able to accurately graph up to three shapes at once with no noticeable lag. Performance will also be measured by the accuracy of where shapes are placed and how accurate their size is.
3. **User Experience and Effectiveness** - an user experience effectiveness is best measured with real users. We will measure this with NPS (net promoter score) to judge how much users enjoy using EasyMath. A successful NPS is high. We will also create a number of tasks for users to complete with EasyMath and judge how long the study participant takes to complete it. Success in this case will be completing the task in a reasonable amount of time. Reasonable will be better defined when EasyMath is completed and we are able to determine what a reasonable amount of time is by using EasyMath.

Milestones

Week	Milestone Description	List of Responsibilities
1 (4/1 - 4/5)	Introduction and tutorials	**Matthew**: Tutorial **Esteban**: Tutorial **Jin An**: Tutorial **Tanuj**: Tutorial
2 (4/8 - 4/12)	Tutorials (learning the basics of VR/AR). Present some team ideas to the staff.	**Matthew**: Proposal **Esteban**: Proposal **Jin An**: Proposal **Tanuj**: Blog
3 (4/15 - 4/19)	Creating the team website and blog. Creating the pitch slides and the PRD for our product (EasyMath).	**Matthew**: PRD and presentation **Esteban**: PRD and presentation **Jin An**: PRD and presentation **Tanuj**: PRD and presentation
4 (4/22-4/26)	Setup and installations - for our	**Matthew**: Web interface

	specific needs Hooking up the web interface with the Magic Leap. Rendering of sphere and cylinder on the Magic Leap.	**Esteban**: Shape generation **Jin An**: Shape generation **Tanuj**: Dependencies (install packages, assets, etc)
5 (4/29-5/3)	Implement our basic interactive features: - Stretch - Contract Implement the ability to add multiple shapes	**Matthew**: Interactions **Esteban**: Interactions **Jin An**: Multiple shapes **Tanuj**: Multiple shapes
6 (5/6-5/10)	Finish up last minute MVP features. Show the MVP working product to some early testers to get their input and adapt feedback. Prepare for potential DEMO to the staff or class. Make sure our core product is performing as intended.	**Matthew**: MVP **Esteban**: MVP **Jin An**: MVP **Tanuj**: MVP
7 (5/13-5/17)	Cone and Ellipsoid support on the rendering side of things. At this point we want to make sure we are solid in rendering our main supporting shapes.	**Matthew**: Cone **Esteban**: Cone **Jin An**: Ellipsoid **Tanuj**: Ellipsoid
8 (5/20-5/24)	Cone and Ellipsoid interactions support. Features to resize shape, observe coordinate values, and intercepts between multiple graphs	**Matthew**: Interactions Cone, Ellipsoid **Esteban**: Interactions Cone, Ellipsoid **Jin An**: Features **Tanuj**: Features
9 (5/27-5/31)	At this point we expect being feature complete on the implementation side, so we will	**Matthew**: Stretch goal #1 **Esteban**: Stretch goal #1 **Jin An**: Testing

	start: - Testing our whole system working together and its main components. - Developing the main use cases and getting input for early testers for last minute changes. - Work on the stability of the product.	**Tanuj**: Testing
10 (6/3-6/7)	Work on the Demo Preparation and if time permits spend some extra time developing our stretch goals.	**Matthew**: Stretch goal #1 **Esteban**: Stretch goal #1 **Jin An**: Demo Prep **Tanuj**: Demo Prep
11 (6/10-6/14)	Demo time! At this point we should have a solid product to present and expose to early adopters.	**Matthew**: Demo **Esteban**: Demo **Jin An**: Demo **Tanuj**: Demo

Materials and Budget

Server	$7/month (~$25)	https://www.heroku.com/pricing
Domain	$20	
Video Chat Unity PlugIn	$115	https://assetstore.unity.com/packages/tools/network/webrtc-video-chat-68030

Risks and how they will be addressed

Risk 1:
As student interacts with the shape, the equation on the web interface changes accordingly
Solution:
As we restrict the ways the shape can be resized to common axes and we will see if can use a predefined method/plugins that will generate an equation for each shape. If that doesn't work,

then we will just use the values of the shape generated by unity everytime the shape changes and generate the equations by ourselves.

Risk 2:
Allowing multiple shapes interact with each other because of computational expense of rendering multiple shapes.
Solution:
By using an efficient rendering technique, we can limit our exposure to this risk. Otherwise, we can alleviate some of the load by doing some preprocessing on the web server, so the magic leap isn't doing all the computation.

Risk 3:
For the second stretch goal, all the students and teachers see the same world and all of them can make changes in the world.
Solution:
The video chat API we are thinking of using, we don't know if that will allow us to share the same workspace and allow every user to interact with that world. If that doesn't work, we have to implement a way to share a workspace and allow users to interact with it.

Risk 4:
Latency and unreliable network to communicate between the Magic Leap and web interface.
Solution:
We can limit our exposure to this risk by using a reliable network, the UW WIFI network. We do a lot of testing to test all the edge cases to ensure this communication does not go down. This communication is essential because it will be the only place users can input equations. If networking between the web interface and Magic Leap is unreliable then we will create an interface on the Magic Leap to enter equations.