FAMU/FSU College of Engineering

Department of Electrical and Computer Engineering

Concept Selection

Team 304: FPL ATS Training Application

Names:
Alexis Cross
Kaitlyn Gurtner
Kevin Rodriguez
Christopher Sopeju
Max Urscheler

Date: 10/28/2020

Concept Selection

Concept selection is one of the most important steps in the engineering design process, this is where the concepts generated are compared in a structured and data driven manner. The analytical approach to selecting concepts removes bias and ensures that most fundamentally sound concepts are chosen. The following techniques will be used to aid in concept selection: House of Quality, Pugh Charts, and Analytical Hierarchy Process. Each process will be discussed and then implemented with a final selection made based on the results of each process.

Concepts Considered

Based on the concept generation document produced, the possible concepts have been divided up into nine different categories from which to choose from. These categories of concepts are: Testing, Teaching Methods, User Interface, UI Screen Design, ATS Design, Production Method, Code Development, Anti- Problem, and Biomimicry. From these nine different concepts forms we were able to come up with 98 concepts. This process of concept generation was more of a brainstorming method to obtain all the important concepts for the project. This will be of much aid to selecting the "most important" concept as we have already narrowed down the variety of concepts into three high fidelity concepts and five medium high fidelity concepts. This narrowing down was performed to pinpoint the 8 most vital concepts for the implementation of the design and already puts us at choosing from 8 concepts, instead of the 98 generated previously.

House of Quality (HOQ)

The House of Quality is used to ensure that the customer statement is involved in the design process by comparing customer requirements to technical methods used to achieve those requirements. The House of Quality consists of two methods of requirement analysis, namely an Engineering - Customer Tradeoff Matrix and an Engineering Tradeoff Matrix, that are then combined into one figure. The Engineering - Customer Tradeoff Matrix identifies how the customer needs and engineering requirements impact each other, while the Engineering Tradeoff Matrix identifies the interaction between engineering requirements. The matrices are composed of the following elements and their respective definitions: ↑ = positive correlation, ↑↑ = strong positive correlation, ↓ = negative correlation, ↓↓ = strong negative correlation, and 0/blank = no correlation. Furthermore, polarity indicators (+/-) are used to indicate positive and negative desirability respectively. Each matrix will be derived and then assembled together to form the House of Quality using such elements.

The House of Quality (which contains the Engineering - Customer Tradeoff Matrix and the Engineering Tradeoff Matrix) led to several outcomes. Firstly, it showed that many of our Customer Needs and Engineering Requirements have very little impact on each other. Only a few of them cause a positive correlation on a different need or requirement. Secondly, it showed that all of our needs and requirements have positive implications, meaning that they all have values that we want to increase, not decrease.

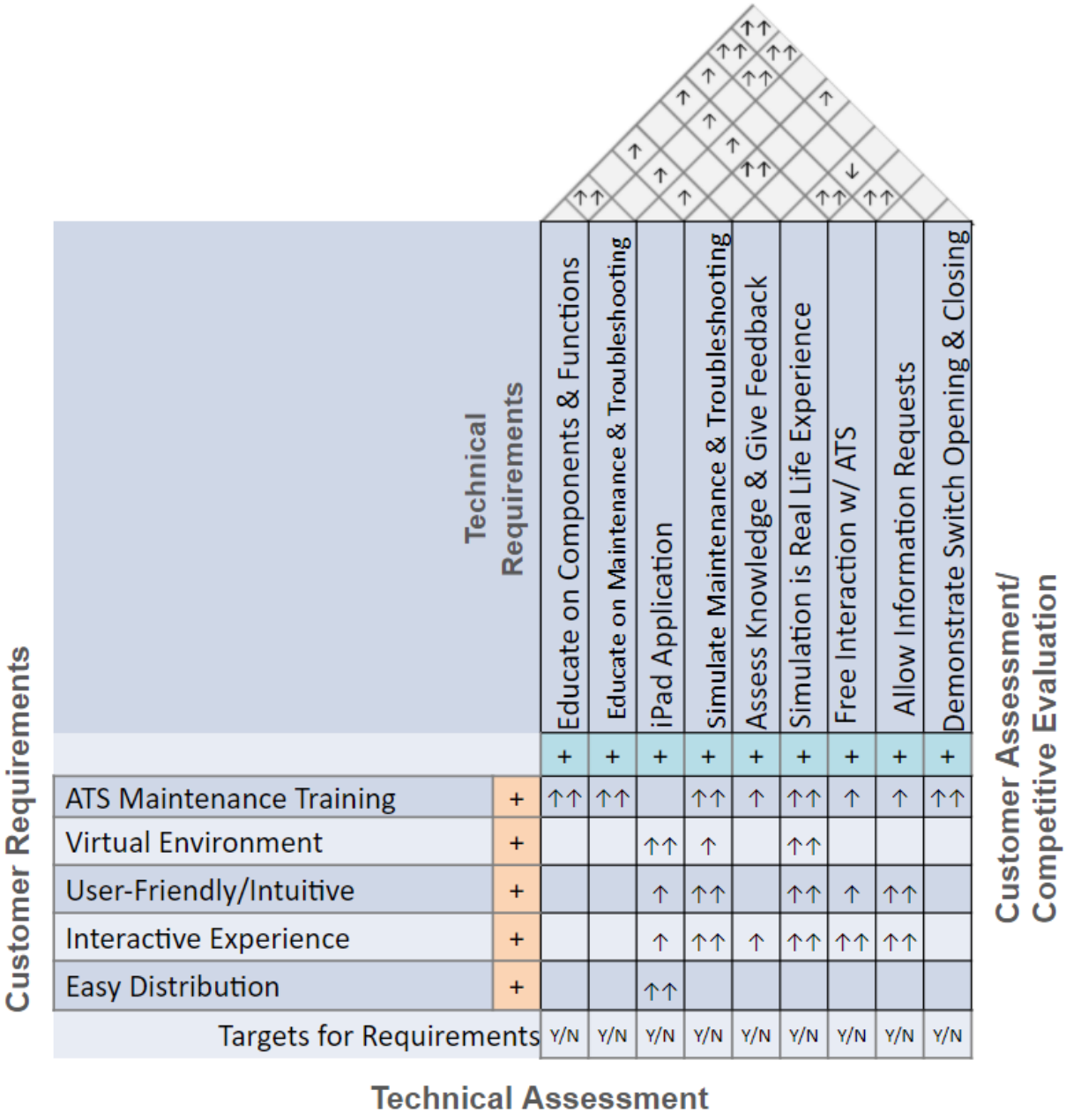
Table 1: Engineering - Customer Tradeoff Matrix

			Engineering Requirements								
			Educate on Components & Functions	Educate on Maintenance & Troubleshooting	iPad Application	Simulate Maintenance & Troubleshooting	Assess Knowledge & Give Feedback	Simulation is Real Life Experience	Free Interaction w/ ATS	Allow Information Requests	Demonstrate Switch Opening & Closing
			+	+	+	+	+	+	+	+	+
Customer Needs	ATS Maintenance Training	+	↑↑	↑↑		↑↑	↑	↑↑	↑	↑	↑↑
	Virtual Environment	+			↑↑	↑		↑↑			
	User-Friendly/Intuitive	+			↑	↑↑		↑↑	↑	↑↑	
	Interactive Experience	+			↑	↑↑	↑	↑↑	↑↑	↑↑	
	Easy Distribution	+			↑↑						

Table 2: Engineering Tradeoff Matrix

			Engineering Requirements								
			Educate on Components & Functions	Educate on Maintenance & Troubleshooting	iPad Application	Simulate Maintenance & Troubleshooting	Assess Knowledge & Give Feedback	Simulation is Real Life Experience	Free Interaction w/ ATS	Allow Information Requests	Demonstrate Switch Opening & Closing
			+	+	+	+	+	+	+	+	+
Engineering Requirements	Educate on Components & Functions	+		↑↑		↑		↑	↑	↑↑	↑↑
	Educate on Maintenance & Troubleshooting	+				↑		↑		↑↑	↑↑
	iPad Application	+				↑		↑			
	Simulate Maintenance & Troubleshooting	+						↑↑			↑
	Assess Knowledge & Give Feedback	+								↑	
	Simulation is Real Life Experience	+							↑↑	↓	
	Free Interaction w/ ATS	+								↑↑	
	Allow Information Requests	+									
	Demonstrate Switch Opening & Closing	+									

Table 3: House of Quality



Pugh Charts

Pugh Charts are a simple methodology to select concepts based on a variety of engineering characteristics. Concepts are compared against criteria with predetermined weights and a reference is established and then the other concepts are compared against the reference in the various criteria using indicators. The indicators will be as follows +1 if the concept is better than the reference, 0 if the concept is equal to the reference, and -1 if the concept is worse than the reference. The weighted scores of each concept are computed to determine if the concepts should be retained or updated (result is positive score), or dropped (result is a negative score). The table is then updated to reflect the results of the first iteration of the Pugh chart and iterations continue until the best concept emerges.

	Code Development	UI Development	Rendering	3D Modeling	App Development
Cost (1)	-1	1	-1	1	0
Ease of Use (2)	-1	-1	-1	0	0
Implementation (5)	1	1	1	0	-1
Versatility (4)	1	0	0	1	1
Module (3)	1	0	-1	1	-1
Score	9	6	-2	8	-4
Continue	No	Yes	No	Combine	

With the Pugh chart above we decided to make cost the least important since most of the software that will be implemented in this project is free. Ease of Use was the next of importance because based on the software that is being used some appear more user friendly than others while others. Versatility is of great importance because the app would be designed for an IPad as requested by the customer, most game development softwares have a very limited amount of operating software they are compatible with. Implementation is of the most importance because one of the available softwares presents advantages with obtaining the customers goals.

Pairwise Comparison

Taken from the Pugh chart in the last section, we used criteria we decided on and compared them to one another based on importance to the project. Cost was given a weight of 1, Ease of Use 2, Implementation 4, and Versatility 3 respectively. We then compared them to one another and we calculated the Geometric and the Criteria weight based on the data listed below to provide each with a score.

	Cost	Ease of Use	Implement ation	Versatility	Module	Geom Weight	Normal Weight
Cost	1	2	5	4	3	1.72	0.238
Ease of Use	1/2	1	5/2	2	3/2	1.50	0.207
Implement ation	1/5	2/5	1	4/5	3/5	1.257	0.174
Versatility	1/4	1/2	5/4	1	3/4	1.35	0.187
Module	1/3	2/3	5/3	4/3	1	1.40	0.194
Scores (Sum)	2.283	4.567	11.4167	9.13	6.85	7.257	1

Analytical Hierarchy Process (AHP)

The Analytical Hierarchy Process is a matrix-based method that compares the different design options in the project by giving each a score. This result is found from the element by element multiplication of the matrix from Pairwise comparison(Wt) and the Criteria weight vector(P). From summing up the scores from every criteria, it seems that Unity/Maya will be the most efficient option to use for the project given the 4 criterias in which we have based our project on. We will naturally use some of the other applications like 3-D Autocad for the model of the ATS to be used for the application which we can import into Unity or Maya, but using Unity/Maya for the creation of the application will be the most ideal option based on scores from the AHP table.

Analytical Hierarchy Process

Ws = C .* W

C =

1	2	5	4	3
1/2	1	5/2	2	3/2
1/5	2/5	1	4/5	3/5
1/4	1/2	5/4	1	3/4
1/3	1/3	5/3	4/3	1

W =

0.238
0.207
0.174
0.187
0.194

Ws =

	Code Development	UI Development	Rendering	3D Modeling	App Development	P(Criteria Weight)
Cost	-0.238	0.238	-0.238	0.238	0	0.238
Ease of Use	-0.207	-0.207	-0.207	0	0	0.207
Implementation	0.174	0.174	0.174	0.174	-0.174	0.174
Versatility	0.187	0	0	0.187	0.187	0.187
Module	0.194	0	-0.194	0.194	-0.194	0.194
Score(Sum)	0.11	0.205	-0.271	0.793	-0.181	1

Avg CV =

Cost	Ease of Use	Implementation	Versatility
0.1573	0.2645	0.4458	0.3594

CI =

Cost	Ease of Use	Implementation	Versatility
1.28	1.245	1.185	1.213

Final Selection

In conclusion based on the charts above in conjunction with the customers needs/requirements and from the Pugh, Pairwise Comparison and AHP charts that using Unity and Maya as the design options is the most logical in this case.

Table 1: Engineering - Customer Tradeoff Matrix

			Engineering Requirements								
			Educate on Components & Functions	Educate on Maintenance & Troubleshooting	iPad Application	Simulate Maintenance & Troubleshooting	Assess Knowledge & Give Feedback	Simulation is Real Life Experience	Free Interaction w/ ATS	Allow Information Requests	Demonstrate Switch Opening & Closing
			+	+	+	+	+	+	+	+	+
Customer Needs	ATS Maintenance Training	+	↑↑	↑↑		↑↑	↑	↑↑	↑	↑	↑↑
	Virtual Environment	+			↑↑	↑		↑↑			
	User-Friendly/Intuitive	+			↑	↑↑		↑↑	↑	↑↑	
	Interactive Experience	+			↑	↑↑	↑	↑↑	↑↑	↑↑	
	Easy Distribution	+			↑↑						

	Unity/Auto-CAD	Unity/Maya	Unreal/Auto-CAD	Unreal/Maya
Cost (1)	-1	1	-1	1
Ease of Use (2)	0	1	-1	0
Implementation (4)	1	0	1	0
Versatility (3)	0	1	0	1
Score	3	6	1	4
Continue	No	Yes	No	Combine

Based on the two charts it supports the decision. Using both Unity and Maya together it meets the customers needs the best because Unity's and Maya's ease of use, cost and versatility outweighs the how well Unreal and Auto-CAD are when it comes to implementing this project. Maya and Auto-CAD are rendering softwares that come from AutoDesk. Both Maya and Auto-CAD can render the ATS in 3D. Auto-CAD will be able to render the ATS with much more detail than Maya but Maya is able to export assets as FBX files that Unity and Unreal can recognize. Auto-CAD can do the same but it will have to be rendered in low resolution and the file will have to be converted for Unity and Unreal to recognize. Unity and Unreal are game design engines both are free to use. Unity is much easier to use and has extensive tutorials to help beginners get into game development while Unreal is the more professional engine that alot of top Game companies used in implementing their game. Unreal will do a better job at implementing the virtual environment and training exercises than Unity but the learning curve on Unreal is so steep that the negatives out way the positives. Unity is a good mix of "Ease of Use" and such a slight drawback when it comes to Implementation that it is the more preferred engine. Plus Unity can develop apps and games for almost any OS that is out there.