Context Analysis & User Try Out Plan // CI504

IA-PENN GROUP
Melissa Burton
Ankit Agrawal
Becky Popelka

Context Analysis

Product Description

The product that the IA-Penn Group evaluated is a paper-based tutorial introducing the software Scratch and instructing a user on how to create a sprite using Scratch. Scratch is a programming language that is designed for everyone to use, from a novice, an expert, child, or an adult. Scratch was created by the Lifelong Kindergarten Group at the MIT Media Laboratory. We have included several screen shots of the user interface at the end of this section. The name Scratch is inspired by the turntablist technique of scratching. This name was chosen due to the nature of the language, where you can re-mix and re-use parts and create something new and innovative from these parts. Scratch differs from most programming languages because instead of relying on hand-written code that involves meticulous syntax checking and typing code, Scratch is visual and is done by dragging command blocks and assembling these blocks for media manipulation. This methodology of drop and dragging command blocks to substitute for typing code was inspired by concepts from EToys and LegoBlocks. Although Scratch can be used by anyone, Scratch was originally designed for novice programmers. Scratch is flexible and can be used to create simulations, animated stories, games, interactive greeting cards, and many other media intensive projects. Scratch is primarily considered an educational programming language since it allows users to explore and experiment with concepts that are used in more diverse and complex programming languages. These concepts that are used in Scratch include relative motion, absolute positioning, image transformations such as rotations or effects, recorded sounds, cell animation, music, and programmable pens. Additional functions include the use of arithmetic functions, comparisons, Boolean logic, conditionals such as if or if-else, loops such as repeat or forever, and event triggers such as allowing an event to happen if the Sprite touches an object.

Scratch has a number of features but fortunately comes with a paper-

based tutorial that provides lessons for individuals desiring to use Scratch. Due to the time constraints of the project, the IA-Penn Group will be evaluating the first lesson of the tutorial. The title of the first lesson is "Lesson 1: Introducing Scratch and Creating Sprite". The purpose of this introductory lesson is to introduce the user to the program Scratch and start the user off by allowing them to create a sprite character. These Scratch lessons are generated by a company called Shall We Learn. This company has generated 15 lessons for using Scratch and also hosts other features such as Scratch Demos, quizzes, Scratch video lessons and some sample projects created in Scratch.

Although Scratch can be used in a classroom, museums, community centers or at home, Scratch is designed to be intuitive and self-directed, and therefore it is integral that any user be able to use the software with little or no instruction from a teacher or any other mentor.

Scratch is open-source and is available for anyone to download, distribute and use for free. Scratch can be used on most operating systems such as Windows, MAC OS X or Linux computers. Scratch is available to download at http://scratch.mit.edu/ . The expectations of the Scratch Designers are that users would be able to reference lessons or tutorials and successfully use the Scratch program with no teacher instruction. The goals of the IA-Penn Group are to evaluate the usability aspects of the lesson by observing test participants work through the lesson.
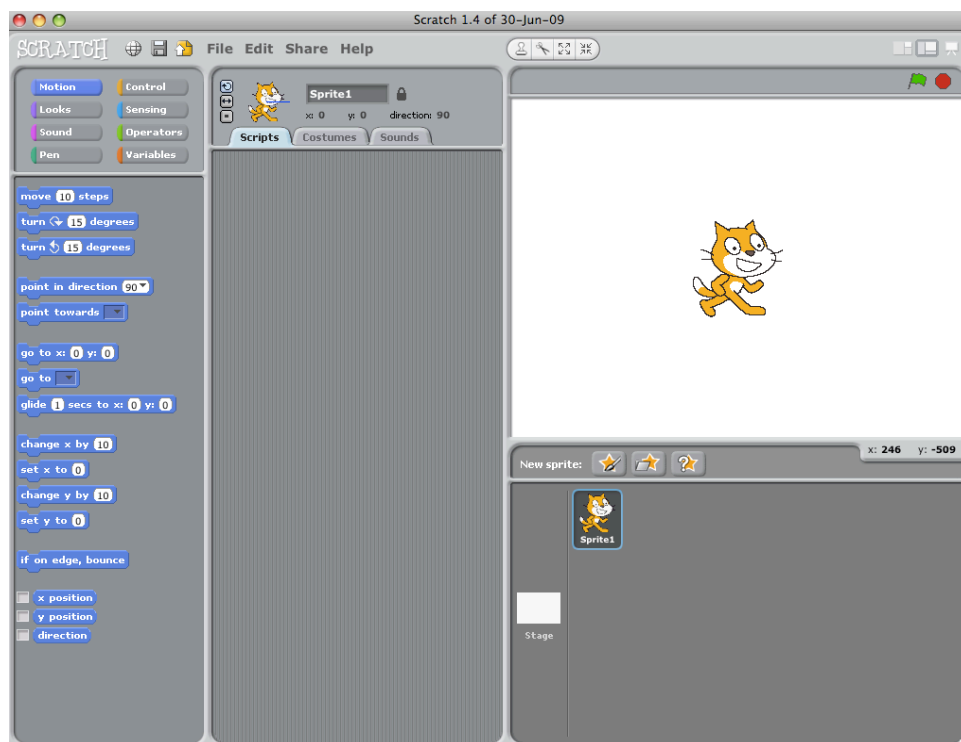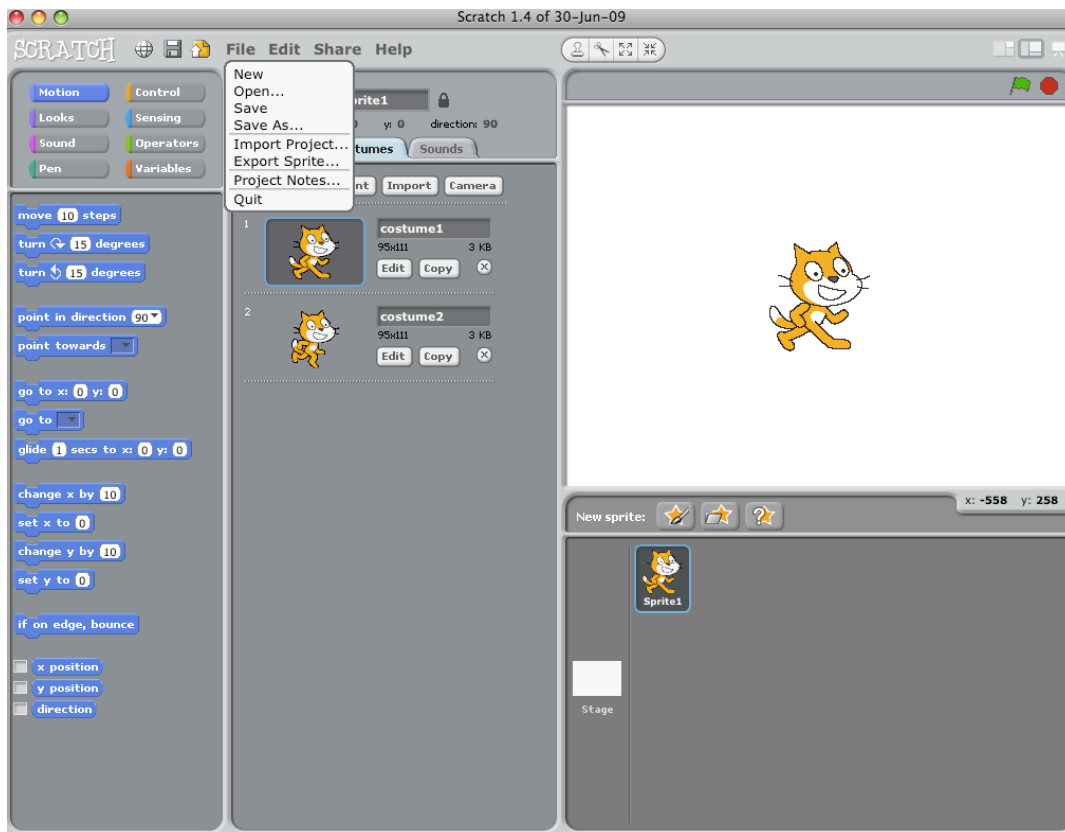

Figure 1: Scratch User Interface

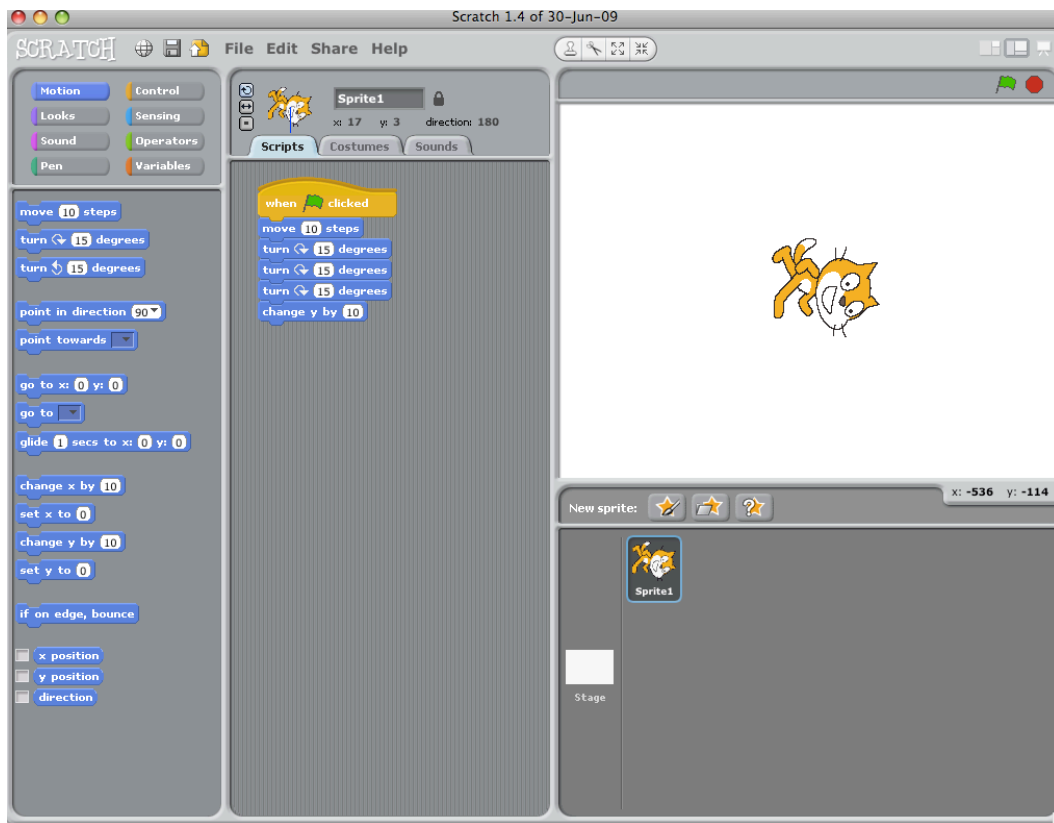*Figure 2:* Scratch User Interface: File drop down menu

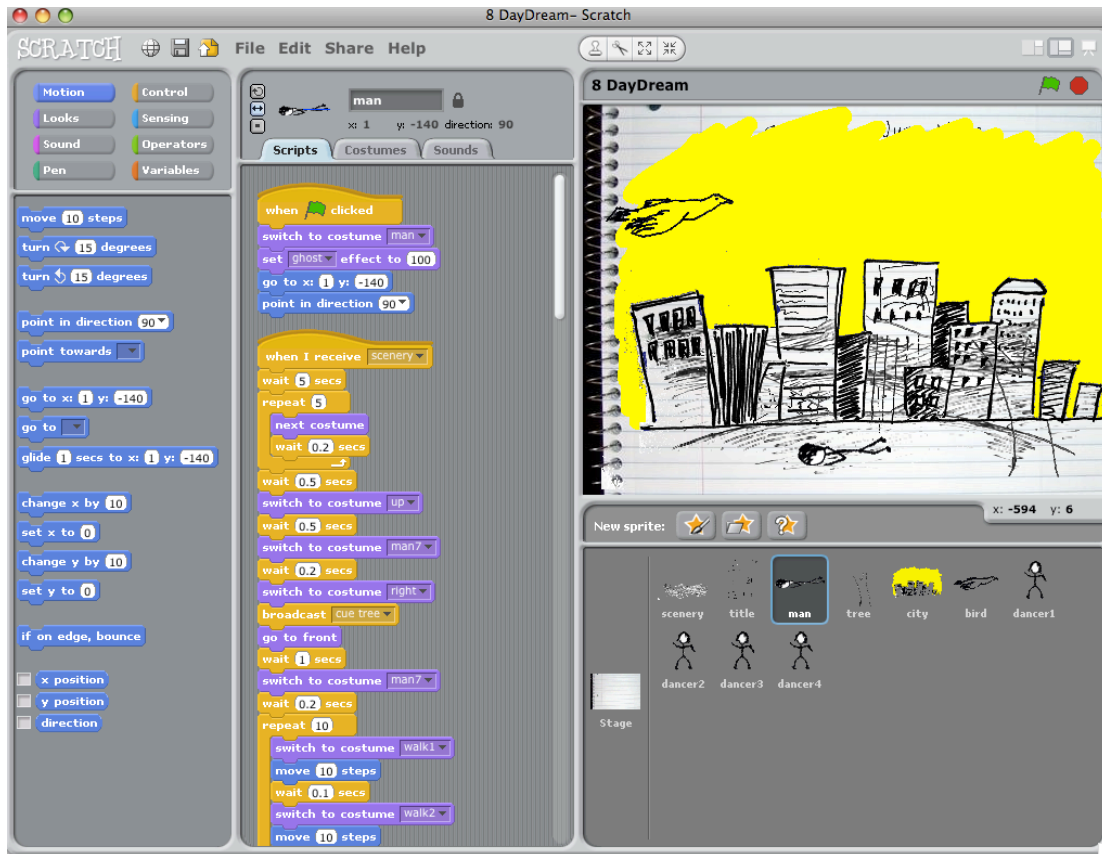*Figure 3:* Scratch User Interface: Animation Steps Added

Figure 4: Scratch User Interface: Example Projects

Description of Target Audience

According to the introduction of the Shall We Learn paper-based tutorial: *Scratch Programming, Lesson 1: Introducing Scratch and Creating Sprite*:

> Scratch is a programming language for all, even for kids. In fact, Scratch, unlike all other programming languages, is designed first and foremost for kids. Because it's designed for kids, it's very easy to learn and use. They can create animations like never before. For older kids or teens, they can create single-level or multi-levels Scratch games.
>
> But Scratch is not just for kids or teens. Teachers and adults can use scratch to create effective education tools such as math quiz, physics simulation and educational videos.
> (Shall We Learn, 2011, p. 1)

Based on this information it can be assumed that the main end user group of the Scratch software product are children and teenagers and a

secondary end user group of teachers and adults. The visual design of the user interface of the software appears to cater to a target audience of children approximately ages seven to eleven years. The design is not likely to appeal to teenagers, although according to the Shall We Learn document the software is aimed at children and teenagers. Trying to appeal to too broad a target audience is often a flaw of educational websites and digital products; it is typically not possible to create instructional materials that appeal to both children and teenagers.

When evaluating the appropriateness of an instructional material designed for a main target audience of children, both in terms of usability and aesthetics it is important to realize that children comprise a group of people at different developmental stages. Co-authors Amy Bruckman and Alisa Bandlow from the Georgia Institute of Technology who contributed chapter 22 in *The Human-Computer Interaction Handbook* look to Piaget's 1970 research to understand child development. This research defined four distinct stages of human development (2003). The first stage is Sensorimotor, beginning at birth and lasting until two years of age. During the Sensorimotor stage children use their sensory perception to build knowledge and thinking skills. The second stage is Preoperational, beginning at two and lasting until around seven years of age. Preoperational children have fleeting attention spans, are able to hold only one item at a time in their memory, are often not able to comprehend abstract ideas and cannot empathize with others. Most preoperational children are not independent readers and are not able to use keyboards. The next stage of development is the Concrete Operational stage lasting from ages seven through eleven. Schneider elaborates on the characteristics of children at this level of development stating, "we see children maturing on the brink of adult cognitive abilities, Though they cannot formulate hypothesis, and though abstract concepts such as range of numbers are often still difficult, they are able to group like items and categorize" (Bruckman & Bandlow, 2003, p. 430). The final Piagetian stage of development is the Formal Operational, beginning at age eleven and continuing into adulthood. At this point children's cognition is similar to that of an adult (Bruckman & Bandlow, 2003).

There are several prerequisite skills needed to operate the Scratch software and work through the Shall We Learn tutorial. The user must be able to: read independently, operate a mouse, operate a keyboard, understand basic software menu functionality, interpret pictorial icons, draw simple figures or images, and have an understanding of the basic concepts and terminology used in the program such as "edit", "costumes" and "controls". Based on the above descriptions of child development stages it is safe to assume that while the main end user group for the Scratch software is "kids", in order to work with the software the users must be at least at the Concrete Operational stage or beyond.

Scratch allows users to create very simple animations or more complex creations such as digital games and therefore users of different ages and skill levels may find the program valuable. However as mentioned before, the visual design of the product caters to younger users. This is not likely to deter adult users, such as the secondary end user group of teachers mentioned in the Shall We Learn tutorial. Teenagers and Pre-teens, on the other hand, are a highly image conscious user group and are likely to shy away from anything they believe appears childish.

Although the main target audience for Scratch is children, initial use of the program and working through the Shall We Learn paper-based tutorial will almost certainly require adult assistance for concrete-operational stage children (ages 7-11). Children younger than 7 years would struggle to use the program at all, while most children 11 and older would be able to work through the tutorial independently. Concrete-operational children would eventually be able to work with the program independently, but will require adult assistance as they are learning the program.

It is expected that users (both adults and children) will approach this instructional material with the attitude that learning about and working with the Scratch software will be a fun experience. The author of the Shall We Learn tutorial states the following in regards to working with Scratch:

> Since I knew about Scratch from a coworker, I have been using Scratch, teaching Scratch, and now writing on Scratch. As you can tell, I just cannot get enough of Scratch. I have two school-age boys and I've been looking for ways to quickly create games and animations to help them learn. Scratch is what I've been looking for and more. It's just a tool so awesome, so fun, and so easy to use and master, that I am sure, once you start, you will be just like me: Can't Scratch Enough! (Shall We Learn, 2011, p.1)

The author's description of their experience using the Scratch software creates an expectation of a fun experience for the user.

Preliminary Evaluation

Scratch is a very interesting product to study and use. It has both strengths and weaknesses. It is very easy to learn and can be used by anyone. The authors claim it is designed for anyone from novice to expert, but we feel that the level of detail and available options are a bit too simple and low in number to get a prolonged serious attention from an expert programmer.

One of the more noticeable negative attributes of the product is its user interface. It is too colorful to be an interface for a serious tool. The colors create the impression that it is designed primarily for kids, but at the same time font size is too small and some of the menu items are difficult to locate. Additionally, a lot of very commonly used features are missing. For example, if you want to create the side view of an animated character there is no option to simply rotate the created character as desired and take snaps to create different angled views. Instead the user will have to do a lot of erasing and redrawing to achieve the same effect. Another bad example of user interface design can be attributed to the 'change color' functionality. If the user wants to change the color of his sprite he has to drag and drop a change color tab from looks onto the scripts window and then change the number in small box according to need. This design concept does not seem to favor either the younger users or expert ones as it is confusing and counterintuitive. The small font size along with a less than ideal color scheme makes the whole thing look unappealing.

Talking about basic limitations on grain level, one can only undo five steps while editing a sprite. Similarly zoom is restricted to 5X. A lot of usability issues bring forward the lack of efforts in designing the tool and give the impression that is just a prototype instead of a serious attempt to build something widely usable and productive.

Quite a few options are not easily understandable and we had to rely on the help function to explain many of the options. One such feature is the option to set the costume center at the bottom left of paint editor which places a dragable cross on the editor window.

A couple of innovative ideas that we liked include the ability to take pictures and record voice options embedded in the tool itself. Clicking on 'camera' tab under costumes, takes a snap and creates a layer with that snap on it. Similarly audio can be easily embedded without the hassle of recording separately and then importing.

To improve the product we would like to suggest a lot of basic changes that do not require much effort. The ultimate objective of this tool/programming language is to be easy and fun to use for both non-programmers and programmers.

Looking at it from a novice or kid's perspective, a lot of quick buttons should be included. There can be different menu items that pop up when the mouse is right-clicked at different locations. For example, if a user wants to change a sprite, there can be a pop up menu that can be controlled by right clicking on the sprite. This will be very helpful as most of the novice users try pressing mouse buttons randomly in attempt to find

an option, which may be hidden somewhere deep in some menu. Other than that quick help should be embedded for functionality tabs that are not widely known. This can be formatted by putting a small question mark next to tab, which on clicking, displays help window for it. This will not only eliminate the need of training before using the tool but also make it self-explanatory and exciting to play with. A few places where this can be done are next to green flag tab and red circle on the top of preview window. The scissor icon on the top, which is actually used as a delete key, looks more like an option to cut something.

As the tool is also supposed to be used by instructors and expert programmers, there should be more professional look to the user interface. As a first thought designing a slightly different interface with more exhaustive menu options like those in other contemporary tools is not a bad idea. Two interfaces and the ability to switch between them is a good choice with a little overhead as it is difficult to capture the interest of kids as well as experts with the same interface.

Overall on first sight tool gives the impression that is it software created for children, however not all aspects of the design are kid or novice user friendly. Although a lot of functionalities do make the product easy to use in some respects, some of the very basic functionalities are missing. We would suggest designers to create two slightly different switchable interfaces to cover wide range of users. As a matter of fact, there can also be an option to customize the interface. This way expert user cannot only configure it as per their need, but instructors can also change it depending on the requirements of students for improved and more efficient learning.

The paper-based tutorial was aimed at introducing Scratch to the users and creating a Sprite using Scratch. It did explain how to create a Sprite using Scratch, but the introduction was rather weak and did not say much about Scratch. It just said how Scratch was good for both kids and teens. There were no concrete points but abstract sentences like "It is very easy to learn and use", "kids can create animations like never before". At the end of the introduction, the author wrote a paragraph about her experiences with Scratch.

The paper-based tutorial was randomly laid out and not designed well. There was text between images that led to the confusion about what part was referring to which image as images were not labeled. Poor image-text coordination made it hard to comprehend and more so when the target audience were kids. It was not thorough as all the tools required were not introduced accurately.

Intended Outcomes

Scratch is designed to make programming engaging, motivating, and informative for everyone. The website for Scratch clearly states that the program is supposed to be usable by novice programmers and is designed to be enjoyable to use. Specifically with lesson 1, the user should be able to know what Scratch is, what the intended objectives of Scratch are, and how to successfully make a Sprite.

Cognitive Objectives:

Start the Scratch program successfully

Operate various tools of the Scratch program such as the "Ellipse Tool", "Erase Tool", "Stamp Tool", "Fill Tool", "Select Tool", "Eyedropper Tool", and "Line Tool"

Understand and operate the various modes of the tools

Create a sprite with the Scratch program

Create multiple views of the Sprite, such as the back, left side, and right side viewpoints of the Sprite.

Identify the Sprite Editor and be able to read the editor.

Save the project file, open and run the project again.

Affective Objectives:

Become more comfortable working with computer programming languages using the turntablist technique

Gain a sense of accomplishment in working with technology to create an interesting and aesthetically pleasing animated character

Enjoy using Scratch to create an animated sprite

Psychomotor Objectives:

Work and control and finger muscles required for mouse and keyboard operation

Improve/ Increase hand-eye coordination required for working with and controlling program tools and functionalities

User's Try Out Plan

Matrix of Objectives and Instruments

Test Objectives	Methods	Questions	Materials
Determine how well the paper-based tutorial is meeting its designated objectives.	Usability Try-out with product and post product try-out questionnaire	1. Is the paper-based tutorial meeting its objective of introducing the Scratch software program? 2. Is the paper-based tutorial meeting its objective of teaching a novice user how to create a Sprite using the Scratch software?	Computer and Scratch software, Post product try-out questionnaire (Likert scale questions and open ended-question)
Rate the level of enjoyment experienced when completing the objectives of the paper-based tutorial.	Usability Try-out with product and post product try-out questionnaire	1. Will novice users enjoy working through the paper -based tutorial as a way of being introduced to the Scratch software? 2. Will novice users enjoy working through the paper-based tutorial as a way of learning how to create a sprite with the Scratch software?	Computer and Scratch software, Post product try-out questionnaire (Likert scale questions and open ended-question))
Ascertain the thoroughness and accuracy of the paper-based tutorial in describing the functionality of the tools available in the Scratch software.	Usability Try-out with product and post product try-out questionnaire	1. Does the paper-based tutorial provide accurate descriptions of the tools needed to create a sprite in Scratch? 2. Does the paper-based tutorial provide a thorough description of the tools needed to	Computer and Scratch software, Post product try-out questionnaire (Likert scale questions and open ended-question)

Test Objectives	Methods	Questions	Materials
Determine how well the paper-based tutorial is meeting its designated objectives.	Usability Try-out with product and post product try-out questionnaire	1. Is the paper-based tutorial meeting its objective of introducing the Scratch software program? 2. Is the paper-based tutorial meeting its objective of teaching a novice user how to create a Sprite using the Scratch software?	Computer and Scratch software, Post product try-out questionnaire (Likert scale questions and open ended-question)
Rate the level of enjoyment experienced when completing the objectives of the paper-based tutorial.	Usability Try-out with product and post product try-out questionnaire	1. Will novice users enjoy working through the paper -based tutorial as a way of being introduced to the Scratch software? 2. Will novice users enjoy working through the paper-based tutorial as a way of learning how to create a sprite with the Scratch software?	Computer and Scratch software, Post product try-out questionnaire (Likert scale questions and open ended-question))
Ascertain the thoroughness and accuracy of the paper-based tutorial in describing the functionality of the tools available in the Scratch software.	Usability Try-out with product and post product try-out questionnaire	1. Does the paper-based tutorial provide accurate descriptions of the tools needed to create a sprite in Scratch? 2. Does the paper-based tutorial provide a thorough description of the tools needed to create a sprite in Scratch?	Computer and Scratch software, Post product try-out questionnaire (Likert scale questions and open ended-question)

Table 1: Matrix of Objectives and Instruments

Matrix Development Process

Resources Used

A few resources were used to help determine the instruments used for data collection and data analysis. The book required for the class, "Handbook of Usability Testing: How to Plan, Design, and Conduct Effective Tests" (Rubin & Chisnell, 2008) was extremely helpful for trying to generate the best ways to analyze the paper-based tutorial used simultaneously with the programming software, Scratch. Many of the methodologies described in the book helped us realize that a post-try out questionnaire was useful and necessary to really understand a user's reaction to the paper-based tutorial while using Scratch simultaneously. The specific chapters from this book that were integral to the decision making process were regarding the processes of conducting a test such as preparing a test plan, setting up a test environment preparing test materials and conducting the test sessions. Another book used was the other required text for the course "Program Evaluation: Alternative Approaches and Practical Guidelines" (Fitzpatrick, Sanders & Worthen, 2011). Our group also made use of the consulting services of our course professor, Dr. Ana Correia, who is also an expert in evaluation.

Generation of Questions

The questions were generated first by coming up with various questions that stakeholders might ask or have concerns about. This original list of questions generated included a wide variety of questions. After the group looked through this comprehensive list of questions and objectives, we realized that we needed to specify the list more towards the paper-based tutorial rather than the Scratch programming software. The decision to make the questions specific to the paper-based tutorial is because our group needs to focus on issues that arise only with the instructional aspects of the project, which only include the paper-based tutorial.

After changing the focus of the questions entirely to the paper-based tutorial, the group decided to find objectives of the paper-based tutorial and aim our questions to focus on those objectives. The paper-based tutorial was designed to help users learn the objectives clarified in that lesson. The lesson specifically states that the aim of the tutorial is to introduce the user to the Scratch programming software and to help the user create a sprite. The first two questions asked the user specifically about how well he or she felt the tutorials were meeting those two designated objectives. Also understanding if a paper-based tutorial is an appropriate way to teach about the Scratch software is also important. Many may be under the assumptions that a paper-based tutorial is an

effective way to introduce the Sprite software and to teach a user how to create a sprite, but realistically that may not be accurate. Therefore users were asked if the paper-based method of delivering content in the tutorial is an effective way to introduce users to the Scratch software and for teaching a novice user how to create a sprite in Scratch.

Furthermore the paper-based tutorial is designed to be enjoyable and encourage the user to use Scratch therefore the paper-based tutorial needs to be fun to use. The paper-based tutorial should also be motivational to use to help ensure the users will want to repeatedly use the lessons provided by the company. For the aforementioned reasons our group will ask the users how much they enjoyed working through the paper-based tutorial as a way of being introduced to the Scratch software and for creating a sprite with the Scratch software.

Another objective of this paper-based tutorial and most tutorials is to present thorough and accurate information. The paper-based tutorial must be thorough and accurate when describing the functionality of the tools available in the software tutorial, and all pictures and diagrams used in the paper-based tutorial must be current and consistent. Therefore the questions were modified to ask about the accuracy of the descriptions of the tools needed to create a sprite in Scratch and how thorough was the description of the tools needed to create a sprite in Scratch.

The tutorial must be designed to be easy to use and the visual layout of the paper-based tutorial must be an effective way of presenting the material. A few questions were created to assess these last two objectives of the paper-based tutorial. The questions are about whether the paper-based tutorial is easy to use for introducing Scratch to a novice user or for creating a sprite in Scratch as a novice user. The last question asks if the paper-based tutorial is effective at presenting the information provided in the paper-based tutorial. All the objectives and questions are summarized in the table above.

Justification of Instruments

There are several validations for selecting a post-product try out questionnaire using a likert scale rating system and an open-ended question in order to evaluate the objectives of the instructional product as defined in the matrix above. The first justification comes from past experiences with usability testing and evaluating various aspects of an instructional experience. One of our group members conducted usability testing in the past and used preference measures as one method of collecting data. A post-test questionnaire was used to collect data on the preference measures that allowed the group member conducting the research to analyze the reported opinions of the participants. An end-of-experience learner questionnaire was employed by two of our group

members to evaluate various aspects of an instructional experience designed by the same group members. This questionnaire asked participants to rate aspects of the experience such as the amount of knowledge gained through the experience and how enjoyable the experience was. The use of a post-product try out questionnaire allowed our group to collect specific data on the opinions of our identified stakeholder group, novice users of the Scratch software. Not only did the questionnaire address specific objectives related to the instructional product, but the use of the likert scale aided our group in determining the severity of potential usability or design issues. The questionnaire will also allowed us to gather suggestions for improvement from test participants.

In chapter sixteen, "Collecting Evaluative Information: Data Sources and Methods, Analysis and Interpretation", the authors of *Program Evaluation: Alternative Approaches and Practical Guidelines,* include a quote from Braverman's 1996 review of surveys in evaluation which notes: "Surveys constitute one of the most important data collection tools available in evaluation." (Fitzpatrick et al., 2011, p. 427). The authors then go on to state: "Surveys are used when the number of sources is too large for cost-effective interviewing and there is a need or desire to have information from many individuals and to analyze it in a quantitative manner." (Fitzpatrick et al. 2011, p.427). Because of time and personnel limitations, the post-product try out questionnaire provided the most efficient method of collecting data on multiple specific objectives related to the quality of the instructional product.

Strategies for Validity and Reliability

One method our group used to help validate our questionnaire is the pilot test. According to Rubin and Chisnell, authors of the *Handbook of Usability Testing*, the research questions developed in the test plan should be used as the basis of the post-test questionnaire (2008, p. 193). Our group used our research questions to create objectives and design our questionnaire, however, the pilot test helped us to know if our research questions (and therefore also the items on our questionnaire) targeted the most important aspects of the instructional product from the viewpoint of the stakeholder.

An additional strategy that we used to support the validity of our questionnaire was employing the expertise gained from our previous experiences with usability testing, the knowledge gained from our course work and textbooks and information gained from consulting with an evaluation expert. Examining the results of the pilot test and applying necessary changes also helped support the reliability and therefore the consistency of the questionnaire. For example, during the pilot test the user expressed that the title of the questionnaire was confusing, so in

order to avoid further confusion in the actual testing the title of the questionnaire was shortened. Additionally, after conducting the pilot test, we realized that in order to elicit specific responses as to user preferences for instructional delivery methods, we needed to revise our questionnaire. We added a question that listed four delivery methods and allows users to rank the methods in order from most to least preferred. This will help us in our data collection.

Pilot Test

The pilot test for our evaluation took place on Thursday, March 24, 2011 at 8:30 AM. Just one person participated as per the requirement. Raj, 45 years old, is a banker with no acquaintance to animation development or related tools in past. The post-product user try-out questionnaire was the instrument used for evaluation along with a short casual discussion about the whole experience. The questionnaire was given electronically through Survey Monkey, free online software to conduct surveys.

The entire activity was completed in 35 minutes. Time was one of our major concerns because of stringent constraints on the time. Everything went smoothly and we obtained some important information about the test. Although it was not required, the user spent sometime understanding the topic of the survey - "Scratch Sprite Tutorial Post Product Try-Out User Questionnaire". So we decided to change it to something simpler, "Sprite Post Try out Questionnaire", to save the user time and confusion.

A close observation during the task revealed that the user was comfortable and found the instructions easy to follow. The user's disinterest to learn the tool any further highlighted that the lesson is not very interesting and intriguing. Bound with the time limits, we decided to keep the lesson the same to obtain more information from future pilot studies.

The questions were easy to understand and follow. The user was very comfortable with the arrangement of questions and categorization. The only concern the participant raised was the similarity in pair of questions under the each sections. However, the participant brought out the point that it is actually a good thing as it will make the user to read the questions carefully before answering.

The user repeatedly talked about other possible ways that could be better alternatives to a paper based tutorial. Therefore to address this concern, we included a question on similar lines where the user can rank their

preferences for different types of tutorials. These statistics will help us to have a more thorough understanding of the shortcomings of a paper-based tutorial.

The post try out discussion was mostly about the overall experience. The user enjoyed the exercise because he watches a lot of animation and was always intrigued as to how animated movies are made. He also expressed that a more interesting lesson could have been completed in the time frame.

Data Collection Process

Our group tested ten users on March 31st. These users were a sample population of the students enrolled in the CI504 course. Each user was provided with a laptop onto which Scratch had been downloaded and installed and a color copy of the paper-based tutorial. The users were asked to work their way through the paper-based tutorial independently and encouraged to "think aloud". The two on-campus group members observed the users during testing. It took the users twenty to thirty minutes to complete the paper-based tutorial, depending on their level of skill and approximately ten minutes or less to complete the online survey. The entire testing session took no longer than forty-five minutes.

Our group assessed the usability of the lesson aimed at teaching users the introduction of Scratch and how to create a sprite using Scratch with the post-product user try-out questionnaire. After users completed the paper-based tutorial they were given the link to the online post-product try out questionnaire. The questionnaire helped us determine how well the users were able to operate the Scratch program with the assistance of the lesson. The post-test questionnaire allowed us to identify problems that the user encountered that may be related to how well the paper-based tutorial met its objectives, the enjoyment experienced while completing the tutorial and using the programming software concurrently, the ease of use of the paper-based tutorial, the accuracy of the tutorial, and/or the visual layout of the paper-based tutorial. The primary focus of this questionnaire was assessing the overall impressions of the user with the product and to pinpoint frustrations or discontent related to the use of the paper-based tutorial with the programming software.

The questionnaire was given electronically through the use of Survey Monkey. Survey Monkey is a free online survey software that allows an individual to upload questions into an electronic survey, disperse those surveys to users electronically via a web browser link, and automatically log the results of the survey and even allows an evaluator tools to view the results of that information graphically or to output those results into an excel sheet or database. The use of Survey Monkey will ensure that

individuals can provide their responses anonymously and directly after completing the paper-based lesson with the use of the programming software. This eliminated any potential awkwardness or lack of anonymity that could occur when a user directly hands a paper-based questionnaire directly to an evaluator. Ensuring the questionnaire was dispersed to users electronically also made the data collection process much easier for the evaluators. This saved the group time because we did not have to manually input data from paper-based questionnaires. The questionnaire took no longer than ten minutes for a user to complete. Almost all the questions were displayed in a likert scale, with one additional question that asked for general comments or suggestions for improvement and one question in which users were asked to rate instructional delivery methods.

Data Analysis Plan

The link to access our post-product try out questionnaire on Survey Monkey is: http://www.surveymonkey.com/s/YDFJYSC. The participant was able to fill out and submit the questionnaire online. The test facilitators were then able to use the analyze results function provided by Survey Monkey to view the percentage of responses that fall under each rating point for each question and an average rating for each question. The online tool also gave test facilitators the ability to create various kinds of charts and export data to Excel for each question.

Using the tools available to us, our group created spreadsheets, charts and graphs in order to help us analyze the quantitative data and recognize patterns in the quantitative data. Taking the mean or average likert scale score for each quantitative question allowed us to know whether users felt positively, neutrally or negatively about each question posed. These average scores provided the answer to our research questions. An average likert scale score of 3.6-5 resulted in a positive answer for a question, a score of 1-2.4 resulted in a negative answer for a question and a score between 2.5 and 3.5 resulted in neutrality. Average score numbers were further analyzed in more detail based on where they fell on the scale, for example, if the average likert scale score for a question is 4.3, it can be reported that the response is generally positive, however, there is still some room for improvement in this area as it did not reach the highest rating possible, or come close enough to be rounded to the highest rating possible, which is 5.

Our group used the quantitative data to assign average likert scale scores to each question, and from the numerical score apply a qualitative value to the question (positive, negative, neutral). The average likert scale scores allowed us to create a list of priorities for product improvement. The highest priority was the item with the lowest score, and the list built itself in this manner.

In order to analyze the open-ended question that asks users to provide suggestions for product improvement our group coded responses. We tracked each different suggestion and tallied repeated suggestions. When the coding was finished we transferred the qualitative data to a quantitative format. From this we were able augment and/ or adjust the priorities list, noting the most frequently mentioned suggestions as the highest priorities and compiling a list in descending order from most mentioned to least mentioned. The final product of our data analysis is a prioritized list of issues or areas of the product that need to be addressed and improved and recommendations or suggestions for improvement.

Final Report

Results and Findings

Questions one through five and question seven in our *Sprite Post-Try Out Questionnaire* used a Likert scale to measure user's opinions of various aspects of the project. Question six asked users to rate four methods of instructional delivery from most to least preferred and question eight was an open-ended question that asked users to offer suggestions for improvement of the instructional product. The following table is a summary of the results of the Likert scale questions which displays a mean Likert scale rating for each question:

Question	Mean Rating
#1 Does the Product Meet its Objectives?	
Is the paper-based tutorial meeting its objective of introducing the Scratch software program?	3.3
Is the paper-based tutorial meeting its objective of teaching a novice user how to create a Sprite using the Scratch software?	3.2
#2 Is Using the Product Enjoyable?	
Will novice users enjoy working through the paper -based tutorial as a way of being introduced to the Scratch software?	3.3
Will novice users enjoy working through the paper-based tutorial as a way of learning how to create a sprite with the Scratch software?	3.2
#3 Is the Product Accurate and Thorough?	
Does the paper-based tutorial provide accurate descriptions of the tools needed to create a sprite in Scratch?	2.3
Does the paper-based tutorial provide a thorough description of	2.4

the tools needed to create a sprite in Scratch?	
#4 Is the Product Effective in Its Method of Delivery?	
Is the paper-based method of delivering the content in the tutorial an effective way to introducing novice users to the Scratch software?	2.9
Is the paper-based method of delivering the content in the tutorial an effective way to teach a novice user how to create a sprite in Scratch?	3.0
#5 Is the Product Design Effective?	
Is the current visual design and layout of the paper-based tutorial an effective way of presenting the information provided in the tutorial?	2.9
#7 Is the Product Easy to Use?	
Is the paper-based tutorial easy to use as a method of introducing Scratch to a novice user?	3.1
Is the paper-based tutorial easy to use as a method of learning to create a sprite in Scratch as a novice user?	2.8

Table 2: Mean Likert Scale Rating of Post Try-Out Questionnaire Responses

In questions 1, 2, 4, and 7, users did not report a significant difference between the two objectives of the instructional product, in all but one of the categories the mean difference between the objective of "introducing Scratch to a novice user" and "teaching a novice user to create a sprite in Scratch" was .1. In one category, Ease of Use, the difference reported between the objectives was .3. In questions 1, 2, 4, users reported that they thought the paper-based tutorial was slightly more effective in "introducing Scratch to a novice user" than "teaching a novice user how to create a sprite in Scratch". In question 7, Method of Delivery, users indicated that they thought the paper-based tutorial was slightly more effective in "teaching a novice user how to create a sprite in Scratch" than "introducing Scratch to a novice user".

Question 3 addressed the accuracy and thoroughness of the product in describing the tools necessary to create a sprite in Scratch. This area received the lowest rating amongst the areas rated with the Likert scale. User's thought the product was slightly more thorough than accurate, resulting in a mean rating of 2.4 for thoroughness and 2.3 for accuracy. These results are not surprising as the paper-based tutorial seems to skip the description of some important steps and features of the product and fails to describe certain navigational steps the user must take to access different areas of the Scratch program. Question 5 addressed the design of the instructional product and revealed that users had a fairly neutral opinion in terms of the effectiveness of the design, which resulted in a mean rating of 2.9.

Overall the mean ratings ranged from 2.3 on the low end to 3.3 on the high end. This can be interpreted as indicating that the users as an overall group felt that the paper-based Scratch tutorial was an instructional product of average quality. While these ratings do not establish that the user group thought the product was excellent or even above average, they also do not establish that user group thought the product was poor or below average. If general terminology is applied to the Likert scale format equating a rating of 1 with poor, 2 with below average, 3 with average, 4 with above average and 5 with excellent, most of the scores land right around 3, either slightly above or slightly below. The one exception to this is in question 3 which addresses the thoroughness and accuracy of the product and in which the mean ratings were closer to 2 (below average) than 3. This would indicate that accuracy and thoroughness is the area in which the product needs the most improvement.

The following two bar graphs compare the results of each of the Likert scale questions:
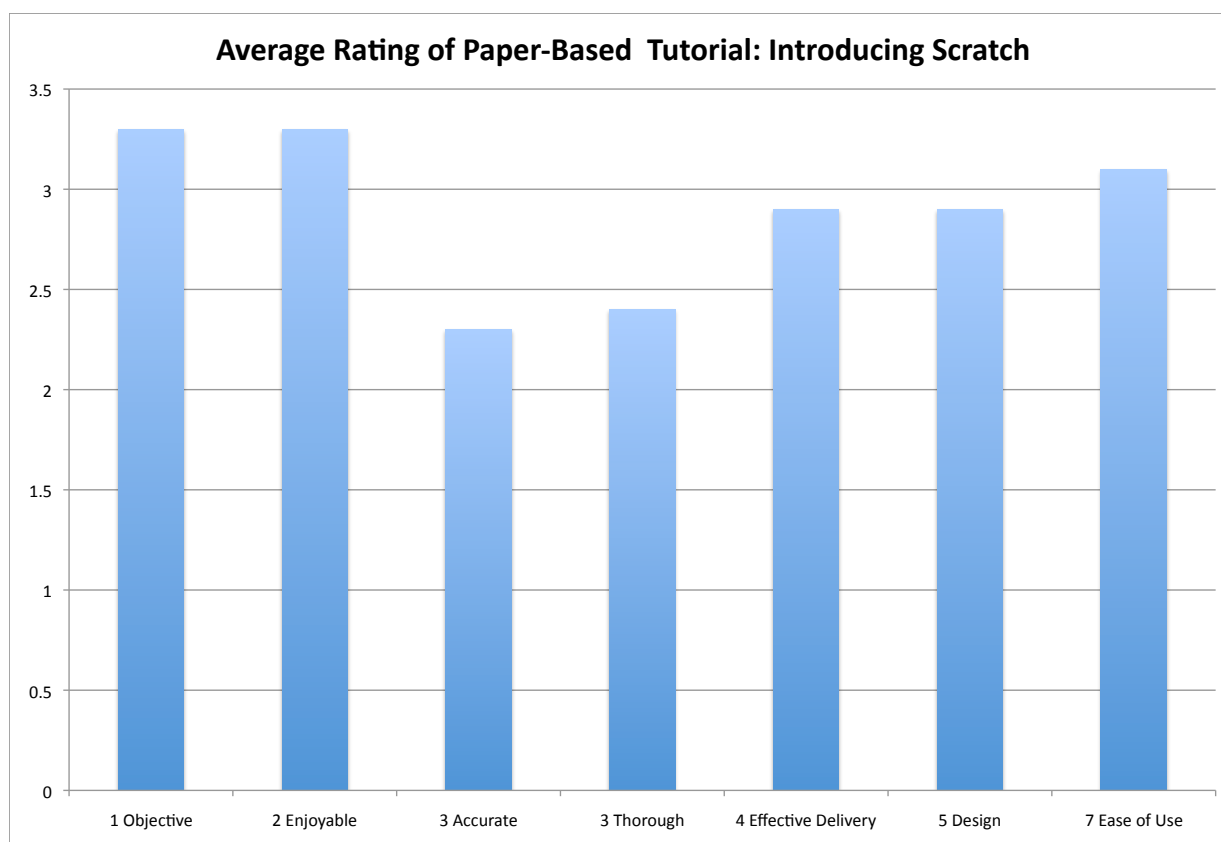

Average Rating of Paper-Based Tutorial: Introducing Scratch

Figure 5: Average Rating of Paper-Based Tutorial: Introducing Scratch

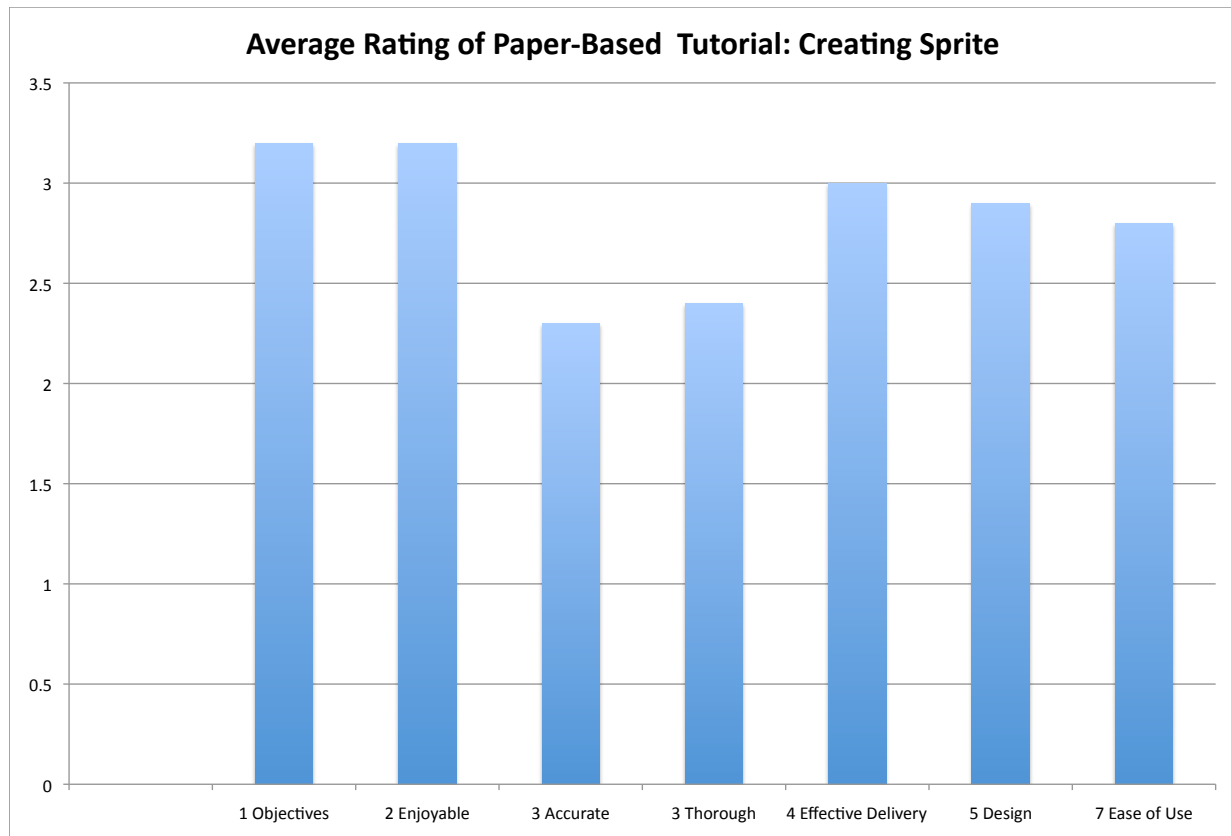**Average Rating of Paper-Based Tutorial: Creating Sprite**

Figure 6: Average Rating of Paper-Based Tutorial: Creating Sprite

Ten individual users rated various attributes of the paper-based tutorial by responding to the online questionnaire. Of those ten, seven assigned ratings that spread across the Likert scale, one assigned all high ratings (4's and 5's) and two assigned all low ratings (1's and 2's).

Because our group was interested in finding out specifically what instructional delivery methods were preferred by our users, we included question 6 in the post-try out questionnaire. Question 6 presented users with four methods of instructional delivery: video tutorial, paper-based tutorial, web-based module, and live demonstration with instructor and asked them to rank the methods beginning with their most preferred and ending with their least preferred. One person skipped this question, and two users answered by assigning multiple delivery methods the same ranking. Therefore, the answers from 7 users for question 6 were analyzed in order to avoid inaccurate data leading to inaccurate analysis. Users reported their most preferred method of delivery as follows: 3 users most preferred a video tutorial, 2 users most preferred a paper-based tutorial and 2 users most preferred a web-based module. Because we were not able to include the responses of 3 of the users, this data may not be significant. The following table displays the results of question 6

Options	1	2	3	4
Video Tutorial	3		3	1
Paper-Based Tutorial	2	1	1	3
Web-Based Module	2	4		1
Live Demonstration with Instructor		2	3	2

Table 3: User Ranking of Instructional Delivery Methods

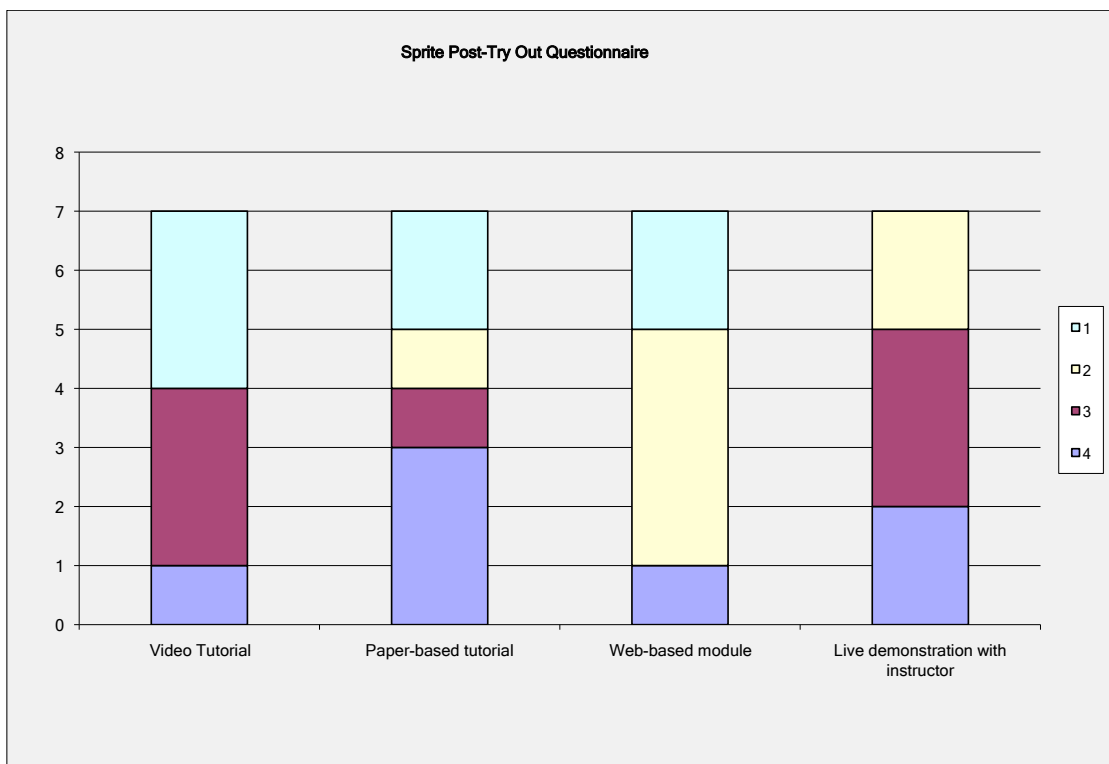The following graph also displays the results of question 6:

Figure 7: graph of user preferences in instructional delivery methods

The last question on the Sprite Post Try-Out Questionnaire was an opened ended question that asked users to provide suggestions for improving the paper-based tutorial. Seven users responded to this question with the following suggestions:

"PDF should be more easy to read with instructions on images. A better task in hand to begin can make things more exciting."

"The directions seem to be for a different version of the software then was loaded on the laptop"

"The paper-based instructions were out-of-sync with the program. I felt that it didn't match the intention of the tutorial. The user was able to learn a few things from just using the software, but in terms of a learning tool, the paper instruction was very ineffective."

"Mine screen of scratch us different for unknown reason; however I manage to follow your tutorial even though at first it quite difficult."

"The version of scratch must be different than the paper tutorial one. This caused some problems."

"Very cool and engaging tool!"

"Did not line up with instructions. Very Challenging. I didn't enjoy the tutorial, was very much looking forward to the end of it. I did enjoy using the software outside of the tutorial - that part was very enjoyable and I'd do it again."

Based on the comments provided by the users, it seemed the biggest issue with the paper-based tutorial was that the tutorial instructions did not line up, or sync correctly with the actual scratch software. This was also the experience our group members had when working through the tutorial. Perhaps the discrepancy in the tutorial and the software is due to the software coming out with a new version and the authors of the tutorial neglecting to update the tutorial. Regardless of the cause, the lack of synchronicity between the paper-based tutorial and the software was a significant cause of frustration for the users.

Recommendations

One of the most pertinent things that need to be improved is that the tutorial needs to address the objectives of introducing the Scratch program and how to create a sprite. Overall the ratings demonstrated that the

tutorial met these objectives fairly, not too poorly or too well. There are two potential ways to address this. The first and possibly best way to address this is to have a tutorial that introduces Scratch and some of the tools. The reasoning behind including a brief introduction to what the Scratch program can do, and some of the tools associated with the Scratch program is so that the user can have a basic understanding of these tools before actually experimenting with the tools. Currently the tutorial blends trying to teach the user how to create and manipulate a sprite, while introducing these tools at the same time. A more efficient way to handle this solution would be to create two separate lessons, one for the introduction of the Scratch programs and the tools, and another for how to create a sprite. Another option that is perhaps less ideal is creating a longer tutorial that includes all the information but the only drawback to this option is that the tutorial may be lengthy.

Best Recommendation: Make two separate lessons. One lesson for introducing the Scratch software and a brief introduction to the tools used in Scratch and a second lesson for creating the sprite.

Alternate Recommendation: Design a longer lesson that covers both objectives, as originally planned, but that may risk being more lengthy and overwhelming.

The second question in the questionnaire asks about how enjoyable it is for a user to learn about the Scratch software and with creating a sprite while working through the paper-based tutorial. On average, the users reported that the tutorial was fairly enjoyable. Overall these findings show that there needs to be additional improvement with making the tutorial more enjoyable to use for users. A way to help create a more engaging tutorial is to change many of the style formats and fonts of the tutorial. Adding more color, changing font types, and improving the overall layout and appearance of the tutorial would certainly make the tutorial more enjoyable to use and aesthetically pleasing. Another recommendation is to change the format of the tutorial to something that could be more interactive for a user and has more flexibility for creating a more enjoyable tutorial. To further demonstrate this point, the fifth question asked the participant if the current visual layout and design is an effective way of presenting the information. The average responses demonstrated that the design was lacking and that regardless changing of the design of the tutorial is relevant and necessary.

Best recommendation: Change the design of the tutorial. Perhaps consider using a different format to increase options and further user's motivation to work with the product.

The lowest rankings from the post product try out questionnaire were from

the questions asking about the accuracy and thoroughness of the questionnaire. The participants ranked the accuracy and thoroughness of the questionnaire to be well below average. The rankings demonstrated that the questionnaire needs significant improvement in terms of adequately conveying the information to the users. The best recommendation for this problem is to have the stakeholder comb through the tutorial and ensure all the steps match up to the current version of the Scratch software. An additional suggestion is to have more usability studies to ensure that the tutorial is thorough and covers everything accurately. The IA-Penn Group expects that a few more revisions would be satisfactory for developing an accurate and thorough questionnaire. This part also ties into the questions that ask about the ease of use of the paper-based tutorial. The participants reported, on average, that the paper-based tutorial was not easy to use when attempting to create a sprite. This also demonstrates that usability studies are needed to help improve the ease of use of the tutorial, and how information is displayed throughout this paper-based tutorial.

Best Recommendation: Revise questionnaire and perform multiple usability studies to ensure an accurate and thorough questionnaire. Multiple usability studies will also help measure the ease of use of the product, and help with revisions.

Another question the IA-Penn Group asked about this instructional instrument is whether the paper-based tutorial was appropriate for learning Scratch. This question was answered through several questions on the post-product try out questionnaire. The users reported that the paper-based tutorial was slightly below average in terms of its effectiveness on introducing the Scratch program to novice users. The participants also reported that the paper-based tutorial performed about average in terms of its effectiveness with teaching novice students to learn how to create a sprite with the Scratch software. From these reported findings, the IA-Penn Group can make some assumptions about how to improve the effectiveness of the tutorial. The first assumption is that perhaps the stakeholders should consider choosing a different format for the tutorial. Currently the format of the tutorial is paper-based and perhaps having a tutorial available in other formats would address some of these inadequacies the user may have when using a paper-based tutorial. Another option is to improve the design of the paper-based tutorial so that the users may feel more satisfied with the tutorial. Fortunately both of these options can be combined.

Best recommendation: Choose different format, other than a paper-based tutorial, for meeting the objectives of introducing the Scratch software and creating a sprite in Scratch. Also improve the design of this format.

Alternative recommendation: Improve the design of the paper-based tutorial.

As mentioned earlier, the users were asked to rank their preference of formats for delivering this information from a video tutorial, web based tutorial, paper-based tutorial and a live demonstration with an instructor. According to the data collected, the users slightly preferred a video tutorial format for delivering information. The data also demonstrated that a couple of users still preferred the paper-based tutorial. Therefore the stakeholders could either choose to create a video tutorial or focus entirely on improving the paper-based tutorial. The IA-Penn Group would encourage the company to create a video tutorial. The reasoning for this is to ensure all users can easily follow along with the instructions. Following along with a video ensures that all the information is up to date and accurate more so than a paper-based tutorial. A video based tutorial also has more options for increasing motivation. Usually following along with a paper-based format can be very overwhelming for a user, especially if the text is long. Also considering the paper-based tutorial is meant for novices of all ages and all educational backgrounds, a paper-based tutorial assumes that the user can really read, understand and comprehend the text in the tutorial. A video tutorial allows a user to be able to read from subtitles if he or she chooses, listen to voices, or simply follow the visuals of the mouse clicking on certain objects on the screen. This will ensure that the video tutorial is accessible for most and allows a user to follow along at their own pace, and to continually check what the video tutorial is showing and how he or she is doing compared to the video.

Comparison

Other than a few criteria, the results of the user try out turned out pretty much the same as one would have expected based on the product critique. A few user opinions did not trace back to findings in the critique primarily because the critique did not cover those aspects well. For example, there are a couple of questions on how enjoyable the experience was. Users were asked if they thought a novice user would enjoy working through the paper -based tutorial as a way of being introduced to the Scratch software and secondly how enjoyable the experience would be as a way of learning. The tutorial was not critiqued over these questions. Users did not find it too enjoyable or too boring on both the occasions with an average likert scale rating of 3.2 and 3.2.

When it comes to other findings, on most occasions they were positive or negative as expected. The tutorial was aimed at introducing Scratch to the users and creating a Sprite using Scratch. Where the tutorial does explain

how to create a Sprite, the critique found it to be a poor attempt at introducing Scratch as a tool. Somewhat contrary to this, users did not find much difference in the tool's ability to meet either of these objectives with a rating of 3.3 and 3.2 respectively. They in fact found it to be slightly better as an introduction to Scratch than as a lesson to create a sprite. The reason could be the ability to quantify the second objective in a better way as compared to first, because the task at hand demanded an understanding of creating a Sprite. There was no task that questioned the user's knowledge and comfort level with Scratch in general; such a task would have underlined the need for a better introduction.

When it came to accuracy and thoroughness, as expected likert ratings slipped further down to 2.3 and 2.4 respectively. The critique broadly raised questions about the way different tools were introduced. It was not easy to comprehend with poor image-text coordination. Method of delivery is another aspect that was not explored in the critique. The critique restrained itself to critiquing the presented format of the tutorial and never went past to question the very medium used to tutor. Again users did not find the paper- based approach highly effective with a rating less than 3 on both the occasions of effectiveness in introducing and teaching.

Critique was harsh on the design and layout of the tutorial, which was supported by a user rating of 2.9 on the likert scale. Like the question on meeting objectives, the one on the ease of use received similar responses. The user found it easier to use as an introductory text than a tutor.

User suggestions ranged from readability issues to issues with the alignment between the tutorial and the current version of software. The majority found the tool itself to be more engaging and intriguing than the paper -based tutorial. One user even said: " I didn't enjoy the tutorial, was very much looking forward to the end of it." Similar things were hinted by the user who took part in pilot run of the user try out plan.

The testing session was very insightful and helped us learn some important things about the user try- outs. Users were far more excited about learning Scratch than we expected. The level of seriousness was commendable and their comments reflected that they were completely involved in the whole process. In particular, the users were annoyed by the fact that the paper-based tutorial and the software were out of sync. We could have tapped more precisely the sections that were frustrating by slipping in an open-ended question about what section they would like to be improved most. Thus, we realized that incorporating a couple of restricted open-ended questions could have provided very important feedback about the paper-based tutorial.

In conclusion, the critique and user try out findings agreed on the broader terms of tutorial being a weak one. They disagreed to a small extent on the quality of tutorial as an introduction to Scratch.

Team Member Contributions

Product Description	Melissa
Description of Target Audience	Becky
Preliminary Evaluation	Ankit
Intended Outcomes	Melissa, Becky
Matrix Objectives	Melissa, Becky
Pilot Test	Ankit
Data Collection Process	Melissa, Becky
Data Analysis	Becky
Results and Findings	Becky
Recommendations	Melissa
Comparisons	Ankit
Presentation	Melissa, Becky, Ankit
Revisions	Melissa, Becky, Ankit

References:

Bruckman, Amy & Bandlow, Alisa. (2003). Human- Computer Interaction for
   Kids. In Julie Jacko & Andrew Spears (Eds.), *The Human Computer
   Interaction Handbook* (pp. 429-440). Mahwah, New Jersey: Lawrence
   Erlbaum Associates, Publishers.

Fitzpatrick, Jody L., Sanders, James R., Worthen, Blaine R. (2011) *Program
   Evaluation: Alternative Approaches and Practical Guidelines, Fourth
   Edition*. Upper Saddle River, New Jersey: Pearson Education, Inc.

Lifelong Kindergarten Group. (2011) Scratch. Retrieved from
   http://scratch.mit.edu/

Resnick, M., Maloney, J., Monroy-Hernandez, A., Rusk, N., Eastmond, E.,
   Brennan, K., Millner, A., Rosenbaum, E., Silver, J., Silverman, B., and
   Kafai, Y. (2009). Scratch: Programming for All. *Communications of the
   ACM*.

Rubin, Jeffery and Chisnell, Dana. (2008). *Handbook of Usability Testing*.
   Indianapolis, Indiana: Wiley Publishing.

Shall We Learn (2011). *Scratch Programming – Lesson 1: Introducing Scratch
   and Creating Sprite*. Retrieved from http://shallwelearn.com/