Lance Newby
11/19/2009
ITIS-3150 – Latulipe

Java Interface
Initial Design Document

Java Interface Building | Speedy Gram

Objective

This is the design of Speedy Gram, the fastest, free, way to grasp grammar. The objective is to learn how to use a WYSIWYG, drag-and-drop design tool (e.g. Java NetBeans or Mircosoft Visual Studio) to produce a production level, and fully functional interface for the Montessori schools' grammar game, Speedy Gram. The fidelity will be fairly high across the board in informational, interactive, visual, editorial, and branding design for the production of this children's game for learning the eight parts of speech within traditional English grammar: the verb, the noun, the pronoun, the adjective, the adverb, the preposition, the conjunction, and the interjection.

Ultimately we want to incorporated appropriate programming logic into the application, which will allow the system to provide feedback to the children and other users interacting with the game.

Audience

This Java interface design is seeking to be as close as possible to final production level. Therefore, this design of Speedy Gram is ideally targeted for the Montessori schools' teachers, and students learning English grammar. In actually grade school teachers and students worldwide should also be able to use this version of Speedy Gram to continue improving students' success in learning grammar.

Team Setup | Solo

This interface design will be individually designed using Java NetBeans (http://netbeans.org/).

Speedy Gram | Prior Prototype

Having already created an Adobe Flash prototype of Speedy Gram I am able to us it as a decent foundation for this version created with Java NetBeans. Taking the feedback received, combined with my own goals of improving Speedy Gram, I will be able to make this iteration of this Montessori schools' grammar game much more refined than the Adobe Flash prototype.

Branding

The logo and name for this grammar game was determined in the early stages of the Adobe Flash prototype design. The name had to be something that was catchy and very simple to remember. Plus the branding name needed to hopefully relay parts of the overall purpose of the software so that if someone heard the name for the first time they could logically come to a conclusion that the software has to do something with learning grammar in a quick manner. Ergo, Speedy Gram was conceived. After coming up with the name I was able to produce a logo, which then paved the way for the overall layout and color scheme of the game.

Improvements From Flash Prototype

- **Determining correct shape is selected for each word**
  The majority of the logic required to determine which of the eight parts of speech shapes was selected for each individual word in a sentence was solved within the Adobe Flash prototype of Speedy Gram. In this first version of Speedy Gram prototype I was able to include the programming logic for storing the correct answers alone with the user's selected answers held within an Array object. Each time that the user places or removed the part of speech from a word the array element associated with the currently selected word is set to a string paired to that part of speech or set to an empty string, respectively.

- **Scoring**
  As in the prior Adobe Flash prototype, I will included one Array object that stores the correct parts of speech for each word in a sentence and one Array object (as aforementioned), with indexes paired to each word in the sentence, that is updated as the user places a selected part of speech shape on a word. I will incorporate levels (elementary, middle, and high) to allow the program to select age appropriate sentences for each user.

    - Levels = elementary, middle or high
    - Sentences Per Level = 4
    - Total Sentences = (4 sentences per level) x (3 levels) = 12
    - Each Correct Answer = 10 points
    - Bonus Round = All answer correct for 3 out of the 5 sentences

Bonus Round

- The bonus round will be rewarded to users who are able to correctly select the parts of speech for 3 out of 4 sentences.
- The bonus round will occur at the end of completing the 4th sentence.
- The bonus round will be similar to the "You Don't Know Jack" Jack Attack round (All the credits go to Jellyvision and Berkeley Systems - http://www.youdontknowjack.com/). See the following for example:

    - http://www.youtube.com/watch?v=XkzjLWHmXxE

- Words will be displayed in large **gray** letters in the background from the pool of words within each of the 4 sentences for the current playing level.
- Parts of speech will be flashed in **blue**, one at a time, at random positions on the screen and will disappear within one second of showing.
- This will continue until the users (using the pressure sensor Phidget) press the sensor when he sees a matching word / part of speech pair on screen!

    - Each Correct Answer = 100 points
    - Each Incorrect Answer = –50 points

Required Interface Components (How they will be incorporated)

1) **Scrollbar** – to show top player's score history at the end of the game.
2) **Drop-down combo box** – Select your Grade Level.
3) **Set of radio buttons** – Select your Reading Level.
4) **Collection of checkboxes** – Optional demographic information (Sp. Ed, Gender, Race, etc…).
5) **Button** – To start the game, move to the next sentence.
6) **Menu set** – Game menu for Quitting the game, Pausing the game, exiting to the Main Menu of the game, starting a New game.
7) **Hierarchical menu** – Similar to Menu set.
8) **Text entry field** – For player to enter their name at the start of the game. The name will be used for communicating more personally to the user throughout the game and for storing high scores at the end of the game.
9) **Larger text entry box** – For receiving feedback from the user at the end of the game.
10) **Invisible link** – The Speedy Gram logo will be an active link to the Montessori schools website (http://www.montessori.edu/). Ideally this link would go to the official website of Speedy Gram.

Content Fidelity Matrix | Java Interface

Speedy Gram | Java Interface Design
Content Fidelity Matrix

Content	Very Low Fidelity	Low Fidelity	Medium Fidelity	High Fidelity	Very High Fidelity
Information Design				✓	
Interactive Design				✓	
Visual Design				✓	
Editorial Content				✓	
Branding Expression				✓	