Educational Software Project: Spelling Games

Project Statement of Needs

The Mad River School District has identified a need to provide its elementary school students with the opportunity to practice and improve their spelling skills. To fulfill this need, the school district applied for a grant to pay for any necessary software, hardware, as well as training and recently received notification it was awarded the grant! The exciting news is that the funding for the grant is largely non-discretionary, with the goal of developing software using technology that best helps elementary-aged school kids learn spelling. The school district currently has a computer lab in each school, running a collection of Windows PCs and well as Macs. However, the nature of the grant will allow the school district to upgrade the machines – or purchase tablets – per the software requirements.

Your company has been selected to develop Spelling game software for 1st thru 5th graders (6 – 10-year olds.) If this software is successful, then the software for the subsequent grades may also be awarded to your company.

In 14 weeks you will deliver the final product to the school district. Intermediary releases will enable continuous feedback. The students and teachers in the district's classrooms and computer labs will use the final product. The success of this project is critical, as the students badly need to improve their spelling skills before they move to high school and have to do extensive writing.

Instructional Requirements

The software needs to provide the students with an opportunity to practice the Spelling skills learned in class. Here are some possible activities that your software might support:

- Writing and filling in sentences with correctly spelled words.
- Filling in missing letters in words.
- Identifying misspelled words.
- Using correct spelling when identifying objects and expressing ideas.

The complexity of the words and the presentation should be age-appropriate i.e. younger children should get simpler words to spell, their interface should be attractive and engaging, and the presentation and interaction should be simple enough to match their limited comprehension and capabilities. Also, when the students are given a problem to solve, they need to receive feedback as to the correctness of their answer. If some words are misspelled, they should be shown the correct spelling, and there should be a higher probability of the misspelled word appearing on a future test (until it is answered correctly).

The instructional activity should be challenging, but not too difficult. The approach taken in your game needs to be appealing to the young audience.

In order to support the educational aspect of the system, student progress needs to be assessed. To accommodate this need, the program can generate a certificate with the student's progress and the teacher can view a student's long-term progress in the program. To facilitate this, each student must enter and verify his/her name upon starting the gaming session. Upon completing a gaming session (which may consist of more than 1 game), the student can generate a gaming progress. Additionally, the teacher can view a student's progress history, whereby the date of each game is printed along with the games score and percentage of problems answered correctly.

In addition, the teacher also needs to be able to add words and spelling questions to the system. This added feature will allow teachers to maximize the usefulness of the system by increasing the set of possible problems.

User Characteristics

The **primary users** will be 1st – 5th graders (6 - 10-year olds) **students**. Many of the kids in this school district have some basic computer familiarity consisting of the ability to:

- use a computer mouse;
- identify the keys on the keyboard and use them to put together numbers, short words, and their name (but the younger students can't touch-type);
- start the computer and programs;
- click on objects (including icons) on the screen;
- enter information in data fields.

Younger children need to be engaged by an interactive system, rather than just a computer-based lecture and test approach. Child-friendly themes and lively colors (and sounds) will help in this endeavor. It is also important to keep in mind that children this age are still learning to read and write. Their vocabulary is still being developed.

The **secondary users** will be **teachers** of the students. The teachers will have differing levels of computer literacy. While many universities require a computer literacy course for graduation, many teachers earned their degree before such a requirement existed. At a minimum, the teacher will know how to:

- start the computer;
- start the program;
- select an item from a drop-down menu;
- click on icons;
- enter information in data fields;
- select options from a set of radio buttons, set of check boxes, and a drop-down list.

Since teachers are college-educated, they can read and follow instructions. Many teachers use word processing and gradebook programs. As such, the use of such application software will prepare them for the utilization of new computer programs. Some teachers are quit savvy using technology, whereas others are still reluctant.

Both the students and teachers may have had some exposure with tablets, but there are no guarantees, as tablets are currently not provided in schools. However, to support an engaging learning system, the grant funding will allow the purchase of tablets – as well as any necessary training – if your software is tablet-based.

Documentation

Your final project should be submitted with online documentation describing how the software functions.

System Requirements

The Mad River School District has no strict policy with regards to what technology the software must be developed with. Fortunately, the grant funding will accommodate whatever hardware needs your software platform requires, as well as any specific training. You may choose to develop a web-based game, a stand-alone desktop application, or a mobile application that is tablet-based.