Abstract

English practice by competition.

Modular two part system:

* Robust Desktop application containing business framework. Operates Game files, describing the game in play. Can be operated offline.
* Server complement. Supplies online multiuser interaction through a Game. May be used to update the desktop program with additional Games.

Throughout the development of this system, we shall be using the book ‘Living English Structure’ by W. Stannard Allen to extract requirements about the structure of exercises.

Business Case

The system shall be used as a platform of exercises relating to certain teacher (syllabus). It shall consists of the basic game engine (to be thoroughly described later), allowing different teachers to present their way of teaching/practicing a language enveloped in an online multiplayer gaming environment.

An Exercise

At the core, the system is a reservoir of ESL (English as Second Language) exercises, targeting three major classes (Elementary, Intermediate and Advanced).

A subject is introduced through various exercises. We shall be implementing the subject structure as present in our guide book. Subjects are focused and integral, and many target aspects of the language.

An exercise practices a subject, in a set difficulty level, through the questions appeared in it. The structure of a question (and hence the structure of the answer) is determined by the type of exercise. For example, an exercise that requires a learner to “complete the sentence”, will contain sentences with missing words, where an answer is a set of words replacing missing values in the sentence.

Different types of exercises may exist for every subject-class relationship.

The Game

Every exercise is a different content in a game.

Automatic “Dummy” Answer Generation

Dummy answers, may be added (automatically) to a question given the correct word answer. Three possible ways to control creation of such answers, given an answer word:

* Stemming. Use the stem of the word to fetch different matches to the same stem. Control play level by using the information of a *distance* between a match and the answer.
* Letter swapping. Any permutation (length>1) of the group (consonant | vowel, consonant | vowel}. Different permutations may be suitable for different difficulty levels. In letter swapping we may also control the number of letters being swapped.
* Syllable count. Words with a higher count of matching syllables may be considered better candidates.

May all be used integrally.

For example, the (Porter) stem for both *appearance* and *appearances* is *appear*:

I don’t remember my first \_\_\_\_\_\_\_\_ at the AA meeting.

1. Place (fetched randomly)
2. Appearances (received from stemmer)
3. **Appearance**
4. Compulsory (close syllable count)

Combined with letter swapping:

1. Appeerance (vowel swap a->e at index 2 counting only vowels, first is zero)
2. Appearances
3. **Appearance**
4. Appearamce (consonant swap->e at index 3 counting only consonants, first is zero)