CHESTR: Computerized Host Encouraging Students To Review

Jordana Hodges, jwhodge1@uncc.edu | Toni Bloodworth, bloodwortht2@gmail.com |
Brandon Kerr, bkerr@uncc.edu | Virgil Simmons, vmsimmon@uncc.edu |
Morris LeBlanc, msleblan@uncc.edu | Dale-Marie Wilson PhD, dalemarie.wilson@uncc.edu

FCL
Future . Computing . Lab

FCL REU students represent Winthrop University and the University of North Carolina at Charlotte

Introduction

CHESTR—or Dr. Chester—is a virtual game show host designed to assist students, computer science in particular, in reviewing for any textbook exam. The game is currently limited to the subject Introduction to Programming (C++).

Background

CHESTR reviews textbook material with students via a trivia game show in which he prompts the user with questions and records how accurately they respond. The answers are listed in a multiple choice format, but a limited number of "lifelines" can be used to break the question down into a 50-50 problem.

CHESTR and the other three agents were created using Haptek, a 3D human generator and environment. The agents interact with the student using SALT, a speech recognition program, which causes the computer game to be relatively hands-free just like a real game show would be.

After playing the game, users have their data stored into a database, recording how well they did, what their scores were for different chapters, and what questions they missed. Upon returning to the game at a later date, CHESTR presents the student with their previous scores and enables them to see their academic progress.

Research

Our research included several major steps:

- Virtual Humans & Personalities
  - We researched how people interact with each other and therefore how to make our virtual humans act.
  - We discovered that a very large percentage of human body language is expressed through the face.
  - We then decided to concentrate on facial features when defining movements of the virtual agents.
- Virtual Human Implementation
  - We researched the methods of using virtual humans in the environment that we wanted (the game).
  - We found out that multiple virtual agents in the same player is not done very often, and has not ever been done with a game such as ours.
- Speech Recognition
  - We researched the various ways in which users would answer a given multiple choice questions.
  - After polling several people and giving them questions similar to those in our game, we realized there were a wide variety of different answers we had to account for.
  - Unfortunately we were unable to implement such a scale of speech recognition and had to revert back to simple lettering and true/false.
- Database Integrity
  - We researched methods for which to retain integrity when working with the databases that store user information for the game.

Impact

Talk about what you got in an outline:

- Group Work
  - Gained experience working and communicating together in a group
  - Worked with people of various ethnic backgrounds
- Project Management
  - Gained experience managing a group project and dividing workloads
  - Maintained a list of small goals to accomplish each day
- Responsibility
  - Gained experience keeping ourselves on track and not having to speak with an advisor all the time
- Presentations
  - Gained a great deal of experience and comfort in presenting research-related materials

Host CHESTR displays his "all-that-and-a-bag-of-chips" personality

Conclusions

We learned a great deal during the REU experience:

- Virtual Humans
  - Loading multiple agents in one Haptek window
- Speech Recognition (SALT)
- Game Logic
  - Javascript to initiate the game and keep it going
  - HTML frames and passing information
- Databases
  - MySQL to get values from the database
  - PHP to authenticate logins and return information

Future Work

For starters, our future work would most definitely include working on the agent personalities. We did not have as much time as we would have wanted to develop CHESTR and the three assistants. We have already chosen personalities for each person, the next step would be to write a more life-like script and include more easily recognizable animations.

Next, there are definitely still bugs in the game show code. After switching over to HTML-based pages from PHP-based, we lost the ability to easily verify that the user is still logged in as they try to access the pages of the game.

Finally, we would also want to work on speech recognition. To truly make our game a fully interactive experience, we want the user to be able to say the actual word of the answer instead of just the letter or true/false.