1. Abstract

This project, sponsored by Ohio University, is intended to produce a software game which will allow its users to increase their memory recall of faces and names. Its features will include a simple user interface to run the game and display images, as well as a speech recognition component which will allow the user to speak the name of the person displayed in the image into a microphone and the game will decide whether answer is correct and supply feedback, as well as keep score. This game will serve the needs of those with Alzheimer’s, dementia, and memory loss due to advanced age.

2. Introduction

Ohio University’s project to develop a game to improve memory recall is intended to aid elderly patients with Alzheimer’s dementia in improving their ability to correctly recall the names of friends and loved ones. Patients who cannot remember their family and friends’ names often suffer frustration and embarrassment. Therefore, a speech pathologist at the university suggested that a way to improve memory recall is to repeatedly practice remembering names with pictures of important people in the patients’ lives. Therefore, the university’s request is for software to be designed which will act as a game for patients to play with pictures of people/places/things they want to remember, which will help to improve name recollection.

There are several requirements that the sponsor requires the name game software to fulfill. First, it must be installed on a PC. The patients use mostly Microsoft Windows, but even for those who don’t, the software must still be able to be installed. The game must accept digital images and names specific for each user and must also supply a package of images of celebrities and political figures, relevant to elderly people. Also, the user or care-taker must be able to select different lists of names to be worked on in different sets, the game must prompt the user to say a name into a microphone once the image is presented, the game must keep score and time, and it must respond to correct or incorrect answers with auditory and visual feedback. Finally, the software used for voice recognition must be supplied, along with the microphone.

Other requirements that the sponsor has have to do with ease of use for people who have few computer skills and are cognitively impaired. The user interface must be simple and clear. Any written instructions must be made relatively large for users with visual impairments, and any audio feedback must be loud enough for people with hearing loss.

2.1 Background (client and disability)

Many of the people who will make use of this software game have Alzheimer’s Disease or dementia and suffer from severe memory loss induced by the disease. Others do not have Alzheimer’s or dementia but struggle with short-term memory loss due to advanced age. For those who have Alzheimer’s, the cause of the onset of the disease is unclear. There are, however, a couple of main hypotheses which describe possible causes for the damage inflicted on brain cells in patients with the disease. One possible explanation is the deposition of beta-amyloid plaque, which interferes with communication between brain cells. Another possibility is the entanglement of an essential neuronal
protein called tau. In people with Alzheimer’s this protein undergoes alterations and does not function properly, leading to damaged neurons.  

2.2 Purpose of the Project

Ohio University’s project to develop a software game to improve memory recall is intended to aid elderly patients with Alzheimer’s Disease or dementia, or those who suffer from short-term memory loss due to advanced age, by improving their ability to correctly recall the names of friends and loved ones. Patients who cannot remember their family’s and friends’ names often suffer frustration and embarrassment. Therefore, a speech pathologist at the university suggested that a way to improve memory recall is to repeatedly practice remembering names with pictures of important people in the patients’ lives. Therefore, the university’s request is for software to be designed which will act as a game for patients to play with pictures of people/places/things they want to remember, which will help to improve name recollection.

There have also been studies done, such as one entitled “Improvement of Picture Recall by Repetition in Patients with Dementia of Alzheimer Type” in the International Journal of Geriatric Psychiatry, volume 12, which show that the use of imagery that is repeatedly shown to patients suffering from memory loss, while simultaneously requiring them to identify the image, is helpful in improving memory recall. Therefore, inspiration for this software game originates from this evidence.

2.3 Previous Work Done by Others

2.3.1 Products

There are several software memory recall products that are available on the market; however, they are not useful in fulfilling all the specifications determined by the university for a memory recall game. A game called Memory Magic at www.acceleratedlearningmethods.com includes several types of flashing memory games in which pictures are flashed on the screen and the user must either identify them by typing the name in or remember the order of certain images. This is done repeatedly to increase memory recall and memory capacity. The game sells for about $50 and includes several other types of memory games besides the one just described; however, it does not allow for customization of images or voice input. Also, it, like many other memory recall games on the market, is geared toward children and teenagers. Therefore, it will not suffice for the client’s usage.

Another example of memory recall software programs that are available on the market are SharperBrain and SharperMemory by SharperPrograms. They basically have the same functionality as the above described game, but it is geared toward children and adults, with the aim of treating ADD and ADHD. The two games sold together are priced at $740, and have the same shortcomings as the product described above.

2.3.2 Patent Search Results

1 Mayo Clinic at http://www.mayoclinic.com/health/alzheimers-disease/DS00161/DSECTION=causes
2 SharperPrograms: Break Free From ADHD at http://www.sharperprograms.com/
Unlike marketed products, there were no patented memory recall softwares found; however, there were two related patents which describe systems or devices to aid in memory recall for people suffering from Alzheimer’s Disease and other neurological disabilities using images. US Patent 2008/01387783 A1, dated June 12, 2008, and filed by Microsoft Corporation for a system to implement Memory Training Via Visual Journal, describes a method which could improve a person’s cognitive ability by recording certain snapshots of events during the day and then playing them back to the person to enhance his/her recollection of the day’s events. The events can be recorded from the person’s perspective or from a third person’s perspective (e.g., by placing the recording device on top of a dashboard). Image capture is triggered by using physiological or environmental sensors to record stimuli associated, for example, with blood pressure, mood, body temperature, external temperature, location, motion, lighting as well as time-based mechanisms. The system would be comprised of a recording component and also a memory recall component which would play back the “memories” for the person who experienced the events. Figure 1 below is an image of the setup.

US Patent 2005/0203430 A1, dated September 15, 2005, and also filed by Microsoft Corporation for a Recall Device, is very similar to the device discussed above. The same components are involved, as well as the same type of triggering for image capture. The difference is that the physiological or environmental stimuli must be followed by a relatively stable period, measured by an accelerometer, before the image is captured. In that way, clearer images may be captured. An image of the device may be seen in Figure 2 below.

![Figure 1: US Patent 2008/01387783 A1 system setup](http://www.freepatentsonline.com/y2008/0138783.html)

---

While these patents support the notion of using images to enhance memory recall for people with memory impairments, they still do not meet the requirements of the given project, as they are not software games and they do not necessarily exercise the user’s ability to remember names of people through audio input.

2.4 Report Map

In the following pages, three alternative designs as well as an optimal design will be presented for this software. This will be followed by a safety and constraints section, along with a budget, timelines for the production of each device, and an appendix with supplementary information.

3. Project Design

3.1 Alternative Designs

The purpose of this project is to increase the speed and accuracy of name recall in elderly patients by creating an installable software package which will be used as a game. The software will display personal images uploaded by the user and prompt the user to speak the name of the person into a microphone, at which point it will then analyze the spoken word and determine whether or not it is correct. The game will then keep score and time and present the user with statistical feedback at the end. More detailed specifications may be found in the Problem Statement and Proposal. Alternative design one will present the general design for the entire code to be implemented for this project in each of the three designs, along with its particular implementation for voice recognition. The difference in the designs is mostly comprised in how the voice recognition will be implemented, so the designs after the first alternative will only describe different options for voice recognition.

Figure 2: US Patent 2005/0203430 A1 device

4 Free Patents Online : All the Inventions of Mankind at http://www.freepatentsonline.com/20050203430.pdf
3.1.1 Alternative Design 1

**Voice Recognition Implementation:** Voice recognition software development kit (SDK), downloadable as freeware or shareware.

3.1.1a Scope of System

I. User Roles

Two user role types will exist for the use of the application. These will include administrators (aids, caretakers) and users (those with memory-recall difficulty). Administrators will have the ability to program photo list scenarios, create profiles, and run and view created scenarios. Users will have the ability to view created scenarios.

II. Creating a Scenario

Administrators will be able to load person profiles (personal pictures and names) into the memory game application. Photo files in bmp, gif, or jpg compression formats can be uploaded and attached to the person profile. Once saved, the profile will be added to the person list, from which the administrator will be able to choose person profiles to be used in the current game. Administrators will be able to add persons to a scenario and shuffle/reorder them as desired. The person list and scenario list are in a one-to-many relationship, allowing person profiles to be listed several times within a scenario. Figure 3 below is an example of how the interface to create a person profile would look. Figure 4 is an example of the person lists that will be created and from which the administrator will choose profiles to be used in the game.

![Figure 3: Interface to create person profile](image-url)
III. Running a Scenario

All user role types will be able to run a scenario by loading it from its saved file location. Photographs will be shown in the center of the application window. Users will then be able to speak the name associated with the photograph. The program will then indicate to the user whether the selection was correct and then advance to the next photograph until completion.

3.1.1b Architecture

I. Navigation

Upon opening the memory game application, the navigation menu can be used to access any of the application’s functionality. This includes loading scenarios, saving scenarios, accessing help and support and printing lists. Figure 5 is a flowchart which describes navigation through the application for both the administrator and the user.

II. Design Considerations

The goal of the memory game application is to be as accessible as possible. An intuitive user interface and step by step and on the fly accessible user help guide will make the application easy for users to understand and set up scenarios. The division between the user and administrator role types will allow for a simple presentation to the end user of the user role type. This fulfills the goal of making the fundamental features of the application simple and minimalistic for the end user while allowing for it to retain its powerful functionality.
III. Dependencies

An important feature of the [memory game] application is its ability to recognize the voice of the end user and determine correctness. In this design, the requirements to run this system include a microphone and a distributable version of voice recognition software. An advantage of using a voice recognition SDK is that it provides the backbone of the code for the voice recognition that is necessary, and basically allows for the developer’s customization of the given code to the needs of the particular application. Languages often used for this type of development include Visual Basic, C#, C++, and Java.
IV. Class Architecture

Figure 6 below is an outline of the types of classes that will be necessary in the code to implement this design. Each class “handles” a certain function within the game.

3.1.2 Alternative Design 2

**Voice Recognition Implementation:** Dedicated chip

In this alternative design, the software package will incorporate the use of a dedicated chip in order to perform the speech recognition functionality.

A dedicated chip is a separate piece of hardware which is able to be programmed with a given number of words as specified by the manufacturer. The speech processing software is stored on the chip, and can incorporate noise cancelling technology which is beneficial for filtering out background noise. When a word is spoken, the dedicated chip compares it to the previously programmed words, and indicates whether the correct word was received. There are two possible routes to take when using a dedicated chip for speech recognition. The first is a speaker dependent system which has a high accuracy for determining what the user says; however, the tradeoff is it is specifically meant for use with a single individual. The second option is the one which this design will incorporate, and it is speaker independent so that many users can utilize the software; however, there will be decreased accuracy.

A sample circuit obtained from Images Scientific Instruments website\(^5\), as seen in Figure 7, shows a possible layout for use with the software package. In order to train the circuit into understanding the spoken words that are necessary, the user will use a keypad. Each number will correspond to a spoken word. To program the circuit the user will press a number on the keypad and speak the word they wish to train, the spoken word will then be stored under that designated number.

---

Once the initial words have been entered, the user can speak freely, and when the circuit recognizes a stored word, the number corresponding to its saved location will flash.

Sensory Incorporated currently holds the largest market share for dedicated speech recognition, and they have a number of software development packages that come with the software and chips necessary to create projects quickly and efficiently. For this design, one of Sensory’s toolkits would be used in order to develop the software necessary to interact with the dedicated chip.

The main drawback to utilizing a dedicated chip for speech recognition is that for every word, the user must initially enter in a baseline for comparison. Also, the use of a dedicated chip is not multi-user friendly, and there will be a need for extra hardware to be attached to the operating computer instead of a standalone installer. If these drawbacks can be overcome, however, a dedicated chip would be a much simple and relatively inexpensive option.

![Sample Circuit Using a Dedicated Chip](image)

**Figure 7: Sample Circuit Using a Dedicated Chip**

### 3.1.3 Alternative Design 3

**Voice Recognition Implementation: Labview and Vista’s built in voice recognition feature**

One alternative design for the speech recognition game is utilizing National Instruments LabVIEW as the programming language. LabVIEW is programming software that will also allow for development of the name recall game with an intuitive user interface. The speech recognition software that will be utilized will be the built in speech recognition software available on Windows Vista, which is the operating system utilized by the client. LabVIEW has certain audio features which could be utilized to manipulate Vista’s voice recognition into the game. Using LabVIEW will allow the finished program to be exported as an executable that can easily be installed on the user’s computer. As for hardware to go
along with the software, a microphone will be included to accept all user input. Below, Figure 8 displays a sample graphical user interface that could be created in LabVIEW. It is similar to the interface presented in the Proposal.

![Sample LabVIEW GUI for name game](image)

**Figure 8: Sample LabVIEW GUI for name game**

### 3.2 Optimal Design

#### 3.2.1 Objective

The memory game application is designed for practicing facial recognition through memory stimulating exercises. The goal of the application is to improve the user’s ability to recall common names based on visual prompts, typically a photograph. The main consideration of design is to make the program as accessible as possible to users of the User role type. This is due to the anticipated variability in users’ prerequisite knowledge of computers and ability to interact with user interfaces. The design that was chosen to be implemented utilizes Microsoft Window’s Software Development Kit (SDK) with the .net framework. This is a freeware kit that allows developers to use the Window’s environment to create customized applications. Microsoft’s Visual Studio Integrated Development Environment, which may be downloaded for free, will be used to develop the game. This design was chosen as it is the most cost-effective option, while also providing all the necessary means for creating both a simple graphical
interface and a voice recognition component. In general, two roles will exist in this program. These roles are discussed below, and Figure 5, above, describes the tasks available to each role, as they comprise the whole program.

3.2.1a Administrator Role

The default mode of memory application will give access to Administrator and User roles. Administrators will have open, create and update access to the user profile database. Administrators will be able to export and import scenario data and initialize a User role session.

3.2.1b User Role

Once an Administrator has initialized a scenario, the user interface is simplified to show only the items necessary to run the scenario. Users will use their voice or keyboard to input the name associated with the displayed picture at each round. (They will automatically be advanced to the next round upon completion). A score indicator will be provided after all rounds of the scenario have been run and the user interface will revert to reflect the administrator role.

3.2.2 Subunits

3.2.2a System Overview

The application will consist of several subsystems. An intuitive user interface will allow the User to access the full functionality of the application through a menu system. Additionally a profile database will keep track of the Users of the program, their associated persons, scenarios, and photos. A handler class will be used to create files containing information on a scenario or load a scenario file. A speech handler class will be used to enable speech input for names during a User’s scenario session. Figure 9 below is a detailed break-down of all the classes, variables, and methods that will be necessary in the code.
3.2.2b User Interface (Administrator Role)

Upon entering the application, the User will be prompted to create or select a profile. The list of current profiles will be loaded from the profile database. Creating, editing or switching profiles can be accessed at anytime from the top menu bar of the program window. The default interface upon admittance will show a list of recent scenarios, a button to run those scenarios, and an option to create new scenarios. Creating or editing a scenario will bring the user to a list of persons attached to that user’s profile. Persons may be added to the scenario list in a one to many relationship so a person can be used more than once in a single scenario. Randomized scenarios can be created and run on the fly from this screen. Additional persons may be added from within the Administrator role user interface. A panel containing various fields pertaining to a person are available from the edit/create person menu option. A person and photo library are maintained within each user profile database.

3.2.2c User Interface (User Role)

The User session is automatically initialized when a scenario is run. The application window then switches to full screen mode and the first round commences. A round consists of a photo being displayed to the User and the User speaking the correct phonemic pattern into the microphone or typing the correct text into the provided text box. The round is then scored for correctness and a new round is shown until all rounds in the scenario are exhausted. The user interface will then revert to the Administrator role and display statistics. Figure 10 below describes the sequence of events that would occur upon running the game for both the User and the Administrator roles.
3.2.2d Profile Database

The back end of the memory game application consists of a profile database. The profile database will be stored and encrypted in save files and handled through a Data Access Object. Figure 11 below describes this database in more detail.

![Database Diagram](image)

3.2.2e Scenario Save Handler

A specially designed class will be used to create encrypted scenario files. These files will consist of the person list associated with the scenario and any other settings needed to run the scenario. This will enable exporting and importing files between workstations or directories.

3.2.2f Speech Recognition

Speech recognition is an essential characteristic of the memory game application. The program will achieve this by using the system.speech reference of the .net libraries. This reference will import Speechlib SDK 5.3 for Windows Vista operating systems and SDK 5.1 for Windows XP operating systems. A speech handler class will handle all speech interaction containing methods to start and stop listening, identify and compare phonemes, and store phonemes for comparison.

3.2.2g Required Hardware
A properly installed microphone is necessary for the speech components of the memory game application to work. Using the Windows speech training wizard increases the accuracy with which the speech libraries can recognize the User’s voice. The team anticipates the program will have minimal memory cost. Recommended system requirements will likely be 512MB of memory on XP machines and 1GB of memory on Vista machines.

3.2.2h System Dependencies

The memory game application requires Windows XP or Windows Vista. A CD-Rom drive is required for installation. .Net 3.0 libraries must be installed on the system for the application to run voice components properly.

3.2.2i Testing

In general, each of the components discussed will be tested class by class, method by method. The majority of the development of the game will most likely be spent debugging. The speech component will probably require the most testing, simply by writing code to accept certain names and then speaking the names into the microphone to test if they are recognized in the context of the game. Testing with multiple user voices will also be an important component to ensure that the speech component will work well with more than one user.

3.3 Prototype

The final product remains very true to the design described up above. One of the ways in which the game was adapted to control for people with different tones and accents in their voices was to make the confidence interval on the voice recognition adjustable for different people. This may be done in the “Create profile” form. The final product was delivered to the client and the team is still awaiting feedback; however, all the testing done on the game has shown that it runs up to standard and fulfills all requirements. Figures 12-17 below show several of the important forms encountered during operation of the game.
Figure 12: Create profile form

Figure 13: Create person form
Figure 14: Create scenario form

Figure 15: Run scenario form
Figure 16: View statistics form by scenario

Figure 17: View statistics form by person
4. Realistic Constraints

One of the constraints that will be encountered by the memory game application is that the game will be restricted to computers running on Windows Vista or XP. A Mac operating system would require another version of the software, which could be supplied if the client expresses interest in this. Another problem is the accessibility of the software game. The game is being designed to be as user friendly as possible, but since this software’s purpose is aimed at patients with cognitive impairments, especially those with Alzheimer’s and dementia, most of them will not be able to install and run the software without some assistance.

In terms of manufacturability, as this project requires the design of software, there isn’t anything to be physically created, except for the CD-Rom, containing the data to be installed on computers to run the game. Depending on the type of operating system being used, the creation of the installation software may differ.

5. Safety Issues

There are no immediate safety issues that this software package will present. The only piece of physical hardware that is implemented in the design is a microphone unit which will be carefully chosen for its ease of use and quality. Because there will be staff present to help the designated clients perform this game safety does not play much of an issue.

6. Impact of Engineering

This design of a memory game application to enhance memory recall in people with Alzheimer’s or other short-term memory impairments will have a societal impact on those who make use of the software. A human being’s ability to interact with those around him/her is an irreplaceable aspect to life, which, if impaired by cognitive disabilities, can have heart-breaking consequences for those affected by the disability and their loved ones. Being able to recognize the faces of people they love, say their names, and engage them on a level similar to that before their memory impairment began can greatly increase quality of life for those with memory impairments.

On a larger scale, this software design is a small addition to the efforts being made to understand memory loss induced by advanced age and especially by Alzheimer’s Disease and dementia. It uses technology to address the issue of poor memory recall and to improve it through repetitive viewing of images. Hopefully its use will inspire patients, family, friends, and caretakers to believe that there is hope for people with these impairments, that more research is needed to help battle these losses, and that technology can play a helpful role in the future.before their memory impairment began can greatly increase quality of life for those with memory impairments.

On a larger scale, this software design is a small addition to the efforts being made to understand memory loss induced by advanced age and especially by Alzheimer’s Disease and dementia. It uses technology to address the issue of poor memory recall and to improve it through repetitive viewing of images. Hopefully its use will inspire patients, family, friends, and caretakers to
believe that there is hope for people with these impairments, that more research is needed to help battle these losses, and that technology can play a helpful role in the future.

7. Life-Long Learning

During the course of designing this project, new skills have been learned, and old skills have been refined. Traits such as time management, responsibility, teamwork, and leadership have also played a large role during the evolution of the designs. Each one of these traits is vital to working in a real world environment, and by practicing such traits Team 8 has gained a valuable foothold for future endeavors.

Time management played a large role because in order to create a software application a specific timeline needed to be laid out in order to get certain aspects of the project done on time so that future work could be built upon its foundation. Time management is an important skill to learn and refine because it is applied in all kinds of jobs and different situations. Also team work and responsibility are traits which are desired when working in the real world because, in most applications, one individual is not responsible for a project, but a group of individuals is. By learning how to work in a team atmosphere and having the responsibility to get the designated work done, you can apply these traits in a favorable way to future opportunities.

The design of this software game helped to greatly improve the programming skills of the team. Being given the opportunity to become familiar with Windows SDK and the .net framework is an excellent opportunity to add to the team’s computer knowledge, which is very useful in a professional engineering environment.

When researching the way in which this design software package would be implemented, a lot of research needed to be performed and an assessment had to be made about the best way in which to approach the project. These in itself provided valuable experience to the team because being able to research and learn how a system works, and then determining the best way to go about completing the task, is a very practical skill that can be used throughout life.

Overall, the skill sets which were learned and applied during the design process of this project provided valuable experience to the team. By refining such traits as leadership, teamwork, and time management the team was able to work well and stay on task in order to complete the design phase for the project. Also a practical application of learning how a system works and applying it to solve a problem was learned. These skills provide valuable experience for implementing what was learned in the classroom into everyday life, and help provide a base which can be built upon for future careers.

8. Budget

The only thing to be purchased for this project is a Logitech, noise filtering microphone, purchased at around $42. The team has received this money, but may choose to purchase a less expensive microphone.

9. Team Member Contributions

For this project, each team member has been reading a book about C#, the .net framework, and Visual Studio, all of which will be used to program the code necessary for the game. The only supply
that must be purchased is a microphone, which has not been ordered yet. There's no rush with that, however. All the necessary software is freeware and has been downloaded. Together, we have drawn up a design for the game, using flowcharts and UML diagrams, which may be reviewed on our website. Up to now, we've done a lot of work to familiarize ourselves with the C# language, .net libraries, and the Visual Studio programming environment. Recently, we've begun writing some of the classes that we will be implementing in the game, specifically the Person, Profile, ProfileLib, and PhotoLib classes.

10. Conclusion

In conclusion, Team 8 has designed a software game which will, upon completion, enhance the facial memory recall of elderly patients with short-term memory loss. It will involve the use of Visual Studio, the C# language with the .net framework libraries, and speech recognition technology.

11. Acknowledgements

Team 8 wishes to thank our client, Dr. Brooke Hallowel, from Ohio University, as well as Dr. Enderle, James Paolino, and Michael Pella whose help and insight has been very valuable.

12. Appendix

Please use this link to travel to the Microsoft Developer Network Library References for help with C#: