Comparison of Visual Basic Interfaces

On the following page, there are two screenshots of the Visual Basic program. The top screenshot is of the old program, and the bottom one is the one I created. As you can see, there are numerous differences. The most notable is the size of the screens (screenshots are smaller than actual size but scaled properly to illustrate differences). The old program seemed cramped for space, hard to locate things, and there was wasted space on the outside where the screen could have been larger. In my program, I increased the size to the full capacity that could be supported on the computers used in the Academic Support Service's Office. By doing so, I was able to space things out better and add a picture box.

Another very noticeable change is the picture box located in the tutor's information section. The picture box contains the picture of the tutor if an administrator at Academic Support Services has assigned one, by doing so in the Picture Management Form (discussed in detail later in this document).

You can also see that I added a 'View Modules' button that brings up a form showing what modules the current tutor has completed (discussed in detail later in this document).

An issue with the old program was that the user had to scroll through the lists on the left of the screen to find a tutor or class, which are very long lists. As a method to correct this, I added text boxes above the lists. The user can now type in all or part of the name of a tutor or course to jump through the lists instead of just scrolling. There was a VB6 bug that I ran into when creating this though. This bug instead automatically called the list click method if I just changed the selected index. Because of this, I had to make the event handler a double click method instead. In addition to double clicking, the user can just hit 'Enter' if they are typing the tutor's or classes name.

In the old program, the user had to right click on a course in the tutor's list of courses and select 'Add' or 'Delete' to add or delete a course. However, the user was unaware of these capabilities, so I created buttons to do this and added them to the main screen. In my program, I added 'Delete Course' and 'Add Course' buttons next to the list of classes the tutor had.

As you can see across the top of the screen in my new program, I have also added a Help menu that explains to the user how to select tutors and courses, delete and add courses to a tutor, delete all assignments, and printing the quick search. This gives the user a quick reference as to how to perform these tasks.

Old Visual Basic Program

File | Tools

Current Tutors:
- admin, carole
- AMMAN, TERESA
- Anderson, Lauren
- Arnoldussen, Amie
- Baynes, Sarah
- Boushley, Samantha
- Brabender, Elly
- Buckrey, Molly
- Cartier, Justin
- Castro, Gabe
- CHEW, SIRAYA

Inactive Tutors:
- Schneider, Cyle
- Tyson, Melissa
- Wiesley, Catie

Current Courses:
- ACCT 205
- ACCT205
- AMER 235
- AMER 305
- ART 134
- ART 205

Tutor Information

First Name: carole	Suite Number: 1
Last Name: admin	Home Phone: 920-403-3875
ID: 999999	Cell Phone:
Grad Date: Fall 2011	Street: 978 Aldrin
Major: education	City: De Pere
Minor: math	Zip Code: 54115
Advisor: Basak	Address Pref: Use home mailbox

Preferences

☐ Hold

Pref. Hours: -2
Pref Tutees: 5
Num. Semesters: 2
Email: carole.basak@snc.edu

Profile Last Updated: 3/25/2009 | View Schedule | Print Assignment Summary

Tutor Courses

Course Name	Course ID	Professor	Other Prof	Recommendation
Biology	BIOL 111	Rappel		No
Introductory Sculpture	ART 224	Baker, K.	Berken, B.	No
Survey of U.S. Literature 1	AMER 235	Crowley, K.	Crowley, K.	No
American Philosophy	AMER 305	dhd		Yes

Group Sessions

Course Name	Course ID	Group Meeting Times	Num of Students	Professor
Tuesday,4PM	math 114	Algebra & finite	1	

Current Connections

Tutee ID	First Name	Last Name	Course ID	Course Name	Professor

Assign Tutee
- First Name
- Last Name
- ID
- Class ID
- Class Name
- Professor

Assign

Old Visual Basic Program

New Visual Basic Program

File | Tools | Help

Current Tutors
- Admin, Ryan
- AMMAN, TERESA
- Anderson, Lauren
- Arnoldussen, Amie
- Baynes, Sarah
- Boushley, Samantha
- Brabender, Elly
- Buckrey, Molly
- Cartier, Justin

Inactive Tutors
- Acosta, Adrian
- Alvarez, Terra
- Braband, Steve
- Brannon, Caitlin
- Burkhart, Ashley
- Dobihal, Theresa

Courses Available to Tutor
- ACCT 205
- ACCT205
- ART 205
- ART 224
- ART 230
- ART 430

Tutor Information

First Name: Ryan	Suite Number:
Last Name: Admin	Home Phone:
ID: 999999	Cell Phone:
Grad Date: 2007-03-03	Street: My Street
Major: My Major	City: De Pere
Minor: My Minor	Zip Code: 54115
Advisor: Advisor	Address Pref: Use home mailbox

Preferences

☐ Hold

Pref. Hours: 0
Pref Tutees: 0
Num. Semesters: 0
Email: My E-Mail

Profile Last Updated: AWAITING TUTOR I | View Schedule | Print Assignment Summary | View Modules

Tutor Courses

Course Name	Course ID	Professor	Other Prof	Recommendation
Database Techniques/Modelin...	CSCI 330	Anderson, D.	Baker, K.	No

Delete Selected Course

Add Tutor Course

Group Sessions

Course Name	Course ID	Group Meeting Times	Num of Students	Professor

Current Connections

Tutee ID	First Name	Last Name	Course ID	Course Name	Professor	Start Date	End Date

Assign Tutee
- First Name
- Last Name
- ID
- Class ID
- Class Name
- Professor

Assign

New Visual Basic Program

Picture Management Form

The screenshot on the following page is of the new Picture Management Form I added in the new program. The purpose of this form is to give the user the capabilities to upload, delete, and assign pictures, as well as view and delete current assignments.

To upload pictures, the user has to click on 'Upload Picture,' which then opens an open file dialog. This dialog box is set up to automatically open at the 'Tutors' folder created on the C drive. If the user selects a picture, the picture is then added to the list of uploaded pictures and displayed in the picture box to the left. Because the picture has been uploaded to the database, the program deletes the picture from the 'Tutors' folder, providing automatic maintenance.

To delete a picture from the database, the user simply selects a picture and clicks 'Delete Picture'. This removes that picture from the list of uploaded pictures.

To assign a picture, the user must select a picture, select a tutor, and then click 'Assign Picture' to correctly assign one. Once this is done, the picture and tutor the user selected are removed from their respective lists. This prevents the user from assigning one tutor multiple pictures or assigning one picture multiple tutors.

The newly created tutor/picture assignment is added to the Current Assignment list view, and the picture is displayed to the right in the picture box. Now that an assignment has been made for this tutor, selecting this tutor on the main screen will bring up their profile information including their picture that the user just assigned. To view an assignment, the user simply clicks on a current assignment, and the picture is displayed to the right in the picture box. To delete an assignment, the user selects an assignment and clicks 'Delete'.

As you can see on the top, I have also added a Help menu that explains to the user how to select a tutor and picture, upload pictures, delete pictures, assign pictures, and view and delete assignments. This gives the user a quick reference as to how to perform these tasks.

I encountered two large problems when creating this form. The first problem was figuring out how to get the pictures for the office to use. The first idea was to take each tutor's picture with a digital camera and upload them to the main computer. However, doing it this way would force Academic Support Services to buy a digital camera, and it would take a significant amount of time to take the pictures and upload them to the computer. To solve this, we were able to get it set up so that an administrator emails a list of tutors' names and IDs to Campus Card, who then locates the pictures and saves them to the J Drive. An administrator in the office then cuts and pastes the pictures into the 'Tutors' folder on the C Drive. Clicking 'Upload Picture' directs the open file dialog to this folder, where all the tutors' pictures are now stored.

The second challenge I encountered was how to save the pictures to the database. The only way I found to do this was to save the pictures as a BLOB, binary large object, when they are uploaded. By doing this, I was able to successfully save them to the database, but to display them, I had to retrieve them from the database and save them back into a picture file on the host computer. As another form of maintenance, I developed a method to delete these locally saved files when the program is closed.

Help

Pictures

snc-logo-lg-color.gif

Upload Picture   Delete Picture

Tutors

AMMAN, TERESA
Anderson, Lauren
Arnoldussen, Amie
Baynes, Sarah
Boushley, Samantha
Brabender, Elly
Buckrey, Molly
Cartier, Justin
Castro, Gabe
CHEW, SIRAYA
CLANCY, PAT

Assign Picture

Current Assignments

Tutor	Picture
Admin, Ryan	monopoly-guy.jpg

Delete Assignment

Done

Modules Completed Form

The screenshot below is the new Modules Completed Form. The purpose of this form is to record when a tutor has completed Modules, which are tasks, seminars, and videos to further enhance the tutor's abilities. An administrator at the Office of Academic Support Services will be able to input, delete, and view when a tutor has completed modules, and this information is also available on the tutor's website to show them what they have completed.

To input a date, the user selects the desired cell, based on the Module Name and type of date. From here, a calendar will pop up for the user to select a date. Once the user selects a date from the calendar, the selected date will show up in the cell the user clicked. Similarly, if the user right clicks on a cell with a date, the Delete option is displayed, which, if clicked, erases the date out of the cell. Right clicking on a cell without a date will not do anything.

As you can see on the top, I have also added a Help menu that explains to the user how to Add and Delete dates. This gives the user a quick reference as to how to perform these tasks.

The final major addition I added to the new VB program is the ability to print off the quick search results, based on how the user sorted the data. In the old program, this capability was not present, and the administrators felt they needed it as a method to create reports, such as how many students are being tutored for a particular course or professor.

The screenshot below is of the printed results of the Quick Search. The user can now open the Quick Search, sort by the column headers, and print the results in whatever order/priority they need. The example below is sorted by the Tutor's last name, but it could have been printed by any of the columns.

The biggest problem I encountered when programming this calculating when the printer reached the end of a page. In my first trial, the first page printed perfectly, but the rest of the data took up 31 pages with only one record on each page. To fix this, I had to calculate when the printer reached the end of a page, tell the printer to start a new page, and return to the top of the page.

Tutor First	Tutor Last	Tutee First	Tutee Last	Course ID	Course Name	Professor
Lauren	Anderson	Cole	Heeg	Hist 130	hist of Latin America	lamar
Lauren	Anderson	nick	spohn	Hist 130	Hist of Latin America	Lamar
Lauren	Anderson	BRAD	HERMAN	MATH 115	PRECALC	YOUNIS
Amie	Arnoldussen	HOJONG	YOO	IDIS 100	COLLEGE WRITING	LEWIS
Amie	Arnoldussen	Keith	Rennart	math 131	Calc 1	Leiderman
Sarah	Baynes	MATT	SULLIVAN	MATH 114	ALGEBRA & FINITE	ASHLEY
Sarah	Baynes	emily	Zaeske	math 114	Algebra& Finite	Ashley
Sarah	Baynes	lucia	Rueda	math 115	pre calc	Muhs
Samantha	Boushley	AARON	BENSON	MATH 102	MATH	ASHLEY
Samantha	Boushley	samantha	lichtenwald	math 115	Pre calc	Muhs
Samantha	Boushley	MELISSA	STREY	MATH 124	SURVEY OF CALC	POSS
Samantha	Boushley	J	Rap	MATH 132	Calculus + Analytic Geometry 2	Scattergood, W.
Ashley	Burkhart	Rachel	Sieverson	span 102	spanish	ellis
Ashley	Burkhart	jean	clifford	span305	spanish composition	collins
Justin	Cartier	JOSE	SEGOVIA	MATH 124	SURVEY OF CALC	BERKEN
Justin	Cartier	Ian	Klein	math 132	calc 2	Thorson
Gabe	Castro	ALISA	ROSS	ENGL 150	INTRO TO LIT	MACDIARMID
PAT	CLANCY	tim	konfek	acct 205	financial acct	vandenberg
Travis	Conradt	Meghan	moffat-Minter	SSCI 224	stats	Ngo
Travis	Conradt	Svenja	Rauchert	SSCI224	stats	Ngo
James	Cox	HOJONG	YOO	ECON 101	MACROECON	ODORZYNSKI
James	Cox	BOBBY	COX	SOCI 111	ANTHROPOLOGY	HYLAND
James	Cox	JOSH	JOHNSON	SOCI 111	ANTHROPOLOGY	HYLAND
Emily	Drain	ANDREW	MCINTYRE	902141	SPANISH	WELCH
Emily	Drain	rachel	severson	span 102	spanish	ellis
Emily	Drain	AMY	HAYHURST	SPAN 102	SPANISH	WELCH
Emily	Drain	KATIE	THEIL	SPAN 102	SPANISH	ELLIS
Emily	Drain	KATHERINE	BURANT	SPAN 102	SPANISH	WELCH
Emily	Drain	michaela	McCuddy	Span102	Spanish	Welch
Emily	Drain	Cayce	Wright	span102	spanish	ellis
Beth	Holt	cHRIS	WELCH	BIOL 120	GEN BIO	HUNNICUT
Beth	Holt	JONATHAN	HUGO	MATH 132	CALC 2	THORSON
Evan	Hucek	EVAN	HUCEK	905187	INTRO ELEM SPANISH	WELCH
Evan	Hucek	nick	spohn905157	hist 130	hist of latin america	lamar
Evan	Hucek	cole	heeg	Hist 130	Hist of Latin America	Lamar
Evan	Hucek	MEGHAN	COWELL	SPAN 102	ELEMENTARY SPANISH	WELCH
Evan	Hucek	Jean	Clifford	SPAN 305	SPANISH COMPOSITION	COLLINS
Angie	Huening	JAN	SWEGLE	CALC 124	SURVEY OF CALC	COOPER