

# **PFAST**

## **Protein Fluorescence and Structure Toolkit**



**University of Rhode Island**  
**CSC 592 Bioinformatics, Spring 2005**  
**Course Project**

**Chi Shen, Nidhi Bansal, Rajiv Menon, Neelima Guduru, Steve Jaegle**  
*with*  
**Dr. Yana Reshetnyak and Dr. Joan Peckham**

# Table of Contents

Table of Contents .....	2
1. Executive Summary.....	3
2. Introduction.....	4
2.1 Background.....	4
2.2 Goals.....	4
2.3 Requirements .....	4
3. Implementation .....	5
3.1 System Architecture .....	5
3.2 User Functions and Narratives.....	6
3.3 Database Design .....	8
3.4 Website Interaction Design .....	8
3.5 Implementation Plan .....	11
4. Results .....	12
4.1 Accomplishments.....	12
4.2 Implementation Status.....	13
5. Project History .....	15
5.1 People.....	15
5.2 Task Assignments.....	16
5.3 Problems Encountered .....	18
5.4 Lessons Learned .....	18
6. Future Work.....	19
7. References.....	21

# 1. Executive Summary

## Background/Significance

Fluorescence spectroscopy is a traditional method of studying how protein structure varies in different environments, such as temperature and acidity. One of the amino acids in proteins, tryptophan, naturally fluoresces and is highly sensitive to its surrounding environment. Thus, the fluorescence characteristics of tryptophan can be used to predict certain structural parameters of proteins.

## Goals

The purpose of the PFAST project is to apply our knowledge of computer science to aid Dr. Reshetnyak's research in protein fluorescence and structure. We aimed to implement a web interface for existing (offline) analysis programs and a fluorescence database for storing experimental results. Together, this provides other researchers an easier way to use Dr. Reshetnyak's tools and browse the results of past experiments.

## Results

We have implemented a new fluorescence database and a web-interface for both the database and Dr. Reshetnyak's Fluorescence & Correlation Analysis Tool (FCAT). The database allows researchers to browse the fluorescence and relevant structural characteristics of proteins. FCAT allows researchers to decompose the fluorescence spectra into component spectra, which depends on the local environment of the tryptophans in the protein. We did not complete the interface for the Structural and Correlation Analysis Tool (SCAT). This tool derives a set of key structural parameters for the protein, which then can be used to associate the tryptophans to the correct component spectra.

## Future Work

There is much potential for future development of this project. The structural analysis and correlation analysis tools both have not been implemented. Also, the analysis programs, currently written in Visual Basic 6, can be rewritten in C++ or other cross-platform languages to reduce execution time and to eliminate dependence on a particular operating system.

## People

The PFAST team consists of two University of Rhode Island professors, Dr. Yana Reshetnyak (Biophysics) and Dr. Joan Peckham (Computer Science), and five Computer Science graduate students, Chi Shen, Nidhi Bansal, Neelima Guduru, Rajiv Menon, and Steve Jaegle.

## Problems and Lessons Learned

There were several areas in which the team would improve if the project were to be repeated. These areas involve understanding the project requirements better initially, better use of meeting time, fixed weekly meetings, and better distribution of tasks based on skill.

## **2. Introduction**

### ***2.1 Background***

Fluorescence spectroscopy is a traditional method for studying how protein structure varies in different environments, such as temperature and acidity. One of the amino acids in proteins, tryptophan, naturally fluoresces and is highly sensitive to its surrounding environment.

The first step is to obtain fluorescence spectra from the protein. Each spectrum is the sum of the emissions from all tryptophans within the protein. The next step is to break the total spectra into its component parts to determine the locations of the tryptophans in the protein. To do this, Dr. Reshetnyak has developed two algorithms: SIMS and PHREQ [1].

The second step is to find a correlation between this component spectral data and the structural properties of the protein. Dr. Reshetnyak's work has shown that there is a statistically significant correlation between the tryptophan fluorescence characteristics of a protein and some structural characteristics of the protein [2]. For this purpose, Dr. Reshetnyak has written an analysis program using the Statistica, a software statistical analysis program.

### ***2.2 Goals***

The purpose of this project is to apply our knowledge of computer science to aid Dr. Reshetnyak's research in protein fluorescence and structure.

Dr. Reshetnyak has developed some software tools and algorithms for analyzing fluorescence and structural data of proteins. For finding correlation, she also has collected many files which represent data and analysis of previous experiments. Finally, she has written software for performing statistical analysis. The software tools are not in a form that is conveniently usable by other researchers.

Our goals are to improve the format of storing experimental data and to provide other researchers better accessibility to this data as well as her software analysis tools. To this end, we will use database technology for data storage and provide web interfaces for the analysis tools and the database.

### ***2.3 Requirements***

Dr. Reshetnyak specified the following requirements for a successful project:

- The database contains the fluorescence and structural parameters of proteins.
- The fluorescence spectra analysis programs must interface with database.
- The structural analysis programs must interface with the above database.
- Design and implement a new program that performs statistical analysis of the spectral and structural data, based on an existing program.
- All programs and databases must be accessible via the web.
- The database design should accommodate database growth.

## 3. Implementation

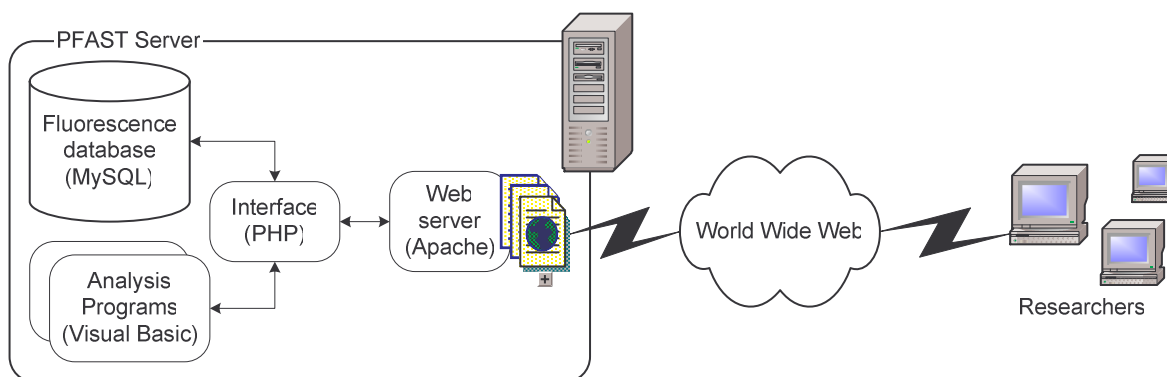
### 3.1 System Architecture

To meet the requirements, the system is composed of many different components. To avoid monetary costs, we have selected a typical collection of free and open-source software for this project:

- Apache web server, for providing the web interface
- PHP scripting, for Apache to connect to the database and analysis programs
- MySQL database, for storing the fluorescence and structural parameters of proteins

One exception to this theme is that the system requires the Windows operating system. This choice is made because the analysis programs can only be compiled for Windows systems. One of the aims of rewriting the analysis programs in the future is to remove this platform dependence.

The overall system architecture is as follows:



### ***3.2 User Functions and Narratives***

Our system has two classes of users: public users and administrators. The following is a list of the functions available to each class of users with brief narratives.

#### **Public User**

- Sign In and Sign Out: On the homepage, registered users may sign in to gain access to the analysis tools. Once signed in, the sign-in form is hidden, and a link for signing out appears.
- Register: From the homepage, unregistered users can follow a link to a registration form. This form collects typical name and contact information as well as a little background on the registrant's research. When the form has been appropriately completed and submitted, the system creates a new user account.
- Browse PFAST Database: From the homepage, all users can follow a link to browse the PFAST database. Users browse the database by specifying a unique PFAST experiment code, a protein, or an author. The system then presents the user with a list of matching experiments. The user can then click on any experiment to see full details.
- Use Fluorescence Analysis Tool: Registered users may follow a link from the homepage to use the Fluorescence & Correlation Analysis Tool (FCAT). First the user specifies if there is just one spectra or more than one spectra to be analyzed. Next, the user has two options: 1) upload a preformatted input file, or 2) fill out some forms to generate an appropriate input file. To upload a file, the user simply selects a file from their computer and proceeds. Otherwise, the user is prompted to provide certain information about the spectra that is being submitted for analysis. The system then generates an appropriate input file. After uploading a file or filling out the forms, the system feeds the input file to the analysis program, which then produces three output files. The system uses one of the output files to generate a graph. The user has the option of downloading the output files and saving the graphs.
- Use Structural and Correlation Analysis Tool: Registered users may follow a link from the homepage to use the Structural and Correlation Analysis Tool (SCAT). First, the user inputs a PDB code. The system then retrieves the PDB file of the corresponding protein from the PDB Databank website. The system then runs the first part of SCAT, which analyzes the PDB file and outputs a set of structural parameters for each tryptophan in the protein. The second part of SCAT analyzes these structural parameters and assigns each tryptophan to fluorescence classes, based on known information about individual tryptophans.

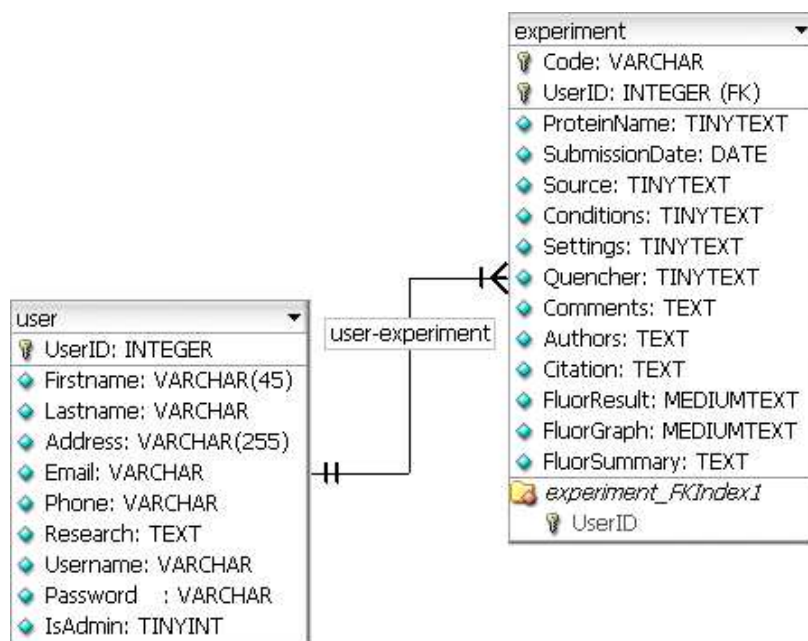
## Administrator

Administrators have access to all functions that public users have. In addition, they can:

- View User Information: From the administrative homepage, an admin user can follow a link to search for a user account. The admin user can search users by name and other reasonable criteria. The system presents a list of matching users, each of which can be clicked on to see the full details.
- Edit User Information: The admin user finds a particular user in the same way as the “View User Information” function. Selecting the “Edit” link for a particular user in the result list leads the admin to a form that allows them to edit any information for that user. Changes made are saved to the database upon submission of this form.
- Delete User: The admin user finds a particular user in the same way as the “View User Information” function. Selecting the “Delete” link for a particular user in the result list leads the admin to a confirmation page. The admin then confirms or cancels the action of deleting the user account. On confirmation, the user account is deleted from the database.
- Save FCAT Results: At the end of the “Use Fluorescence Analysis Tool” function, the admin user has the additional option of saving the results to the database. When the admin user chooses this option, the system presents a form for collecting key information about the experiment. When the form has been completed and submitted, a record of the experiment and its analysis output files are stored in the database.
- Edit Experiment: During the “Browse PFAST Database” function, the admin user has the additional option of editing an experiment. When the admin user chooses this option for an experiment, the system presents a form for editing the details of the experiment. Changes are saved to the database upon submission of this form.
- Delete Experiment: During the “Browse PFAST Database” function, the admin user has the additional option of deleting an experiment. When the admin user chooses this option for a particular experiment, the system presents a confirmation page. If confirmed, the experiment is deleted from the database. Deleted experiments are permanently removed from the database.

### 3.3 Database Design

The database schema of our system is as follows:



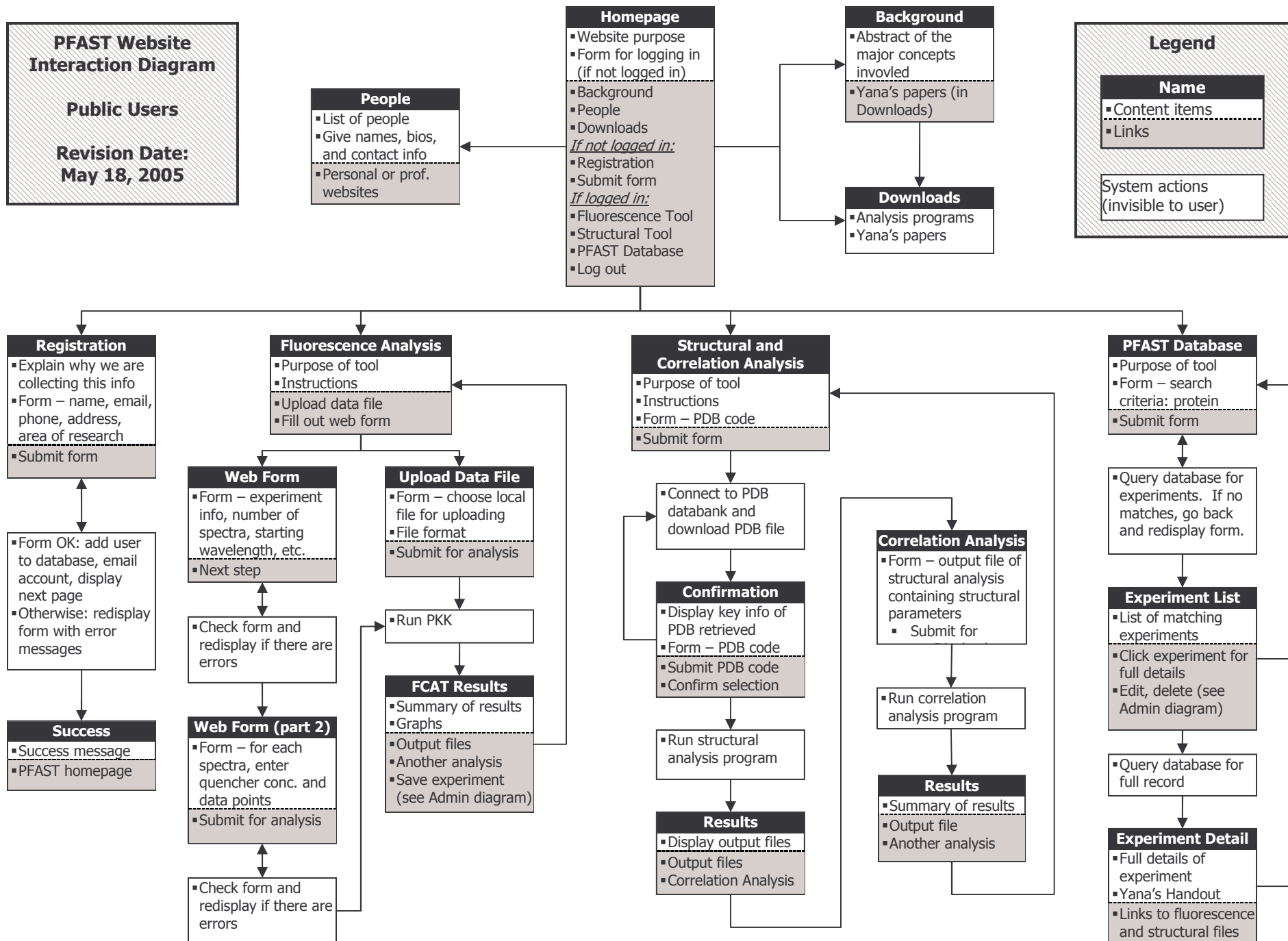
Each saved experiments (FCAT and SCAT results) has a user associated with it. This user is the person that submitted the results to the database. The purpose of most fields are obvious; additional notes for certain fields follow:

- **user:Password** – This field is encrypted using a one-way function, i.e. it cannot be decrypted, and provides some measure of protection in case the database is compromised.
- **user:IsAdmin** – Administrator accounts have a value of 1 in this field, otherwise the value is set to 0.
- **experiment:Code** – This field contains an identifying “PFAST Code” for each experiment, acting much like the PDB code for identifying proteins.
- **experiment:FluorXXX** – These three fields contain the output files saved from FCAT results.

### 3.4 Website Interaction Design

The following two diagrams summarize the content and links of all pages of the web interface to the system. It also shows the sequence of these pages as well as system actions for all functions described in the “User Functions and Narratives” section.





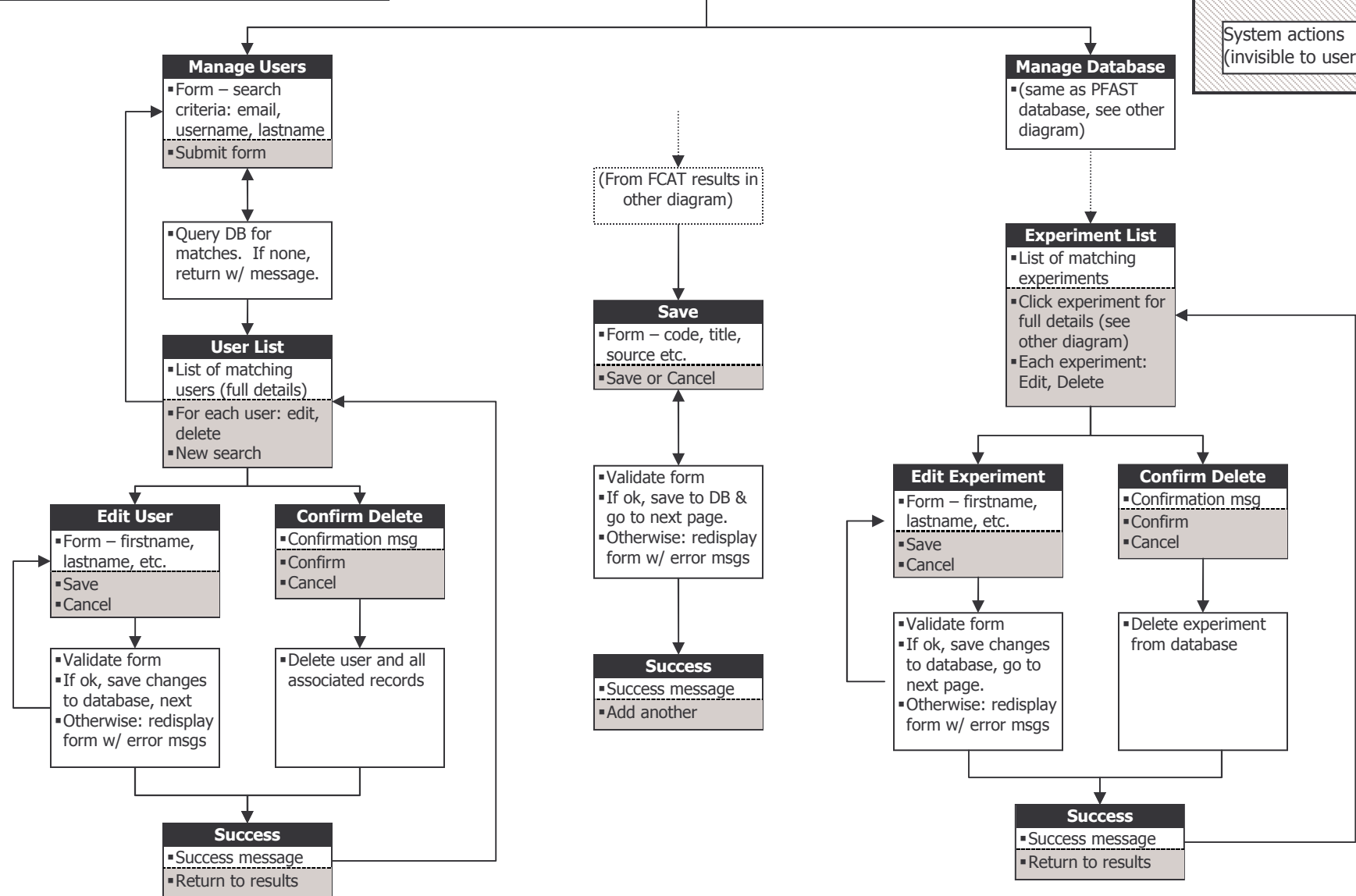
**PFAST Website Interaction Diagram**

**Administrator (Yana)**

**Revision date: May 18, 2005**

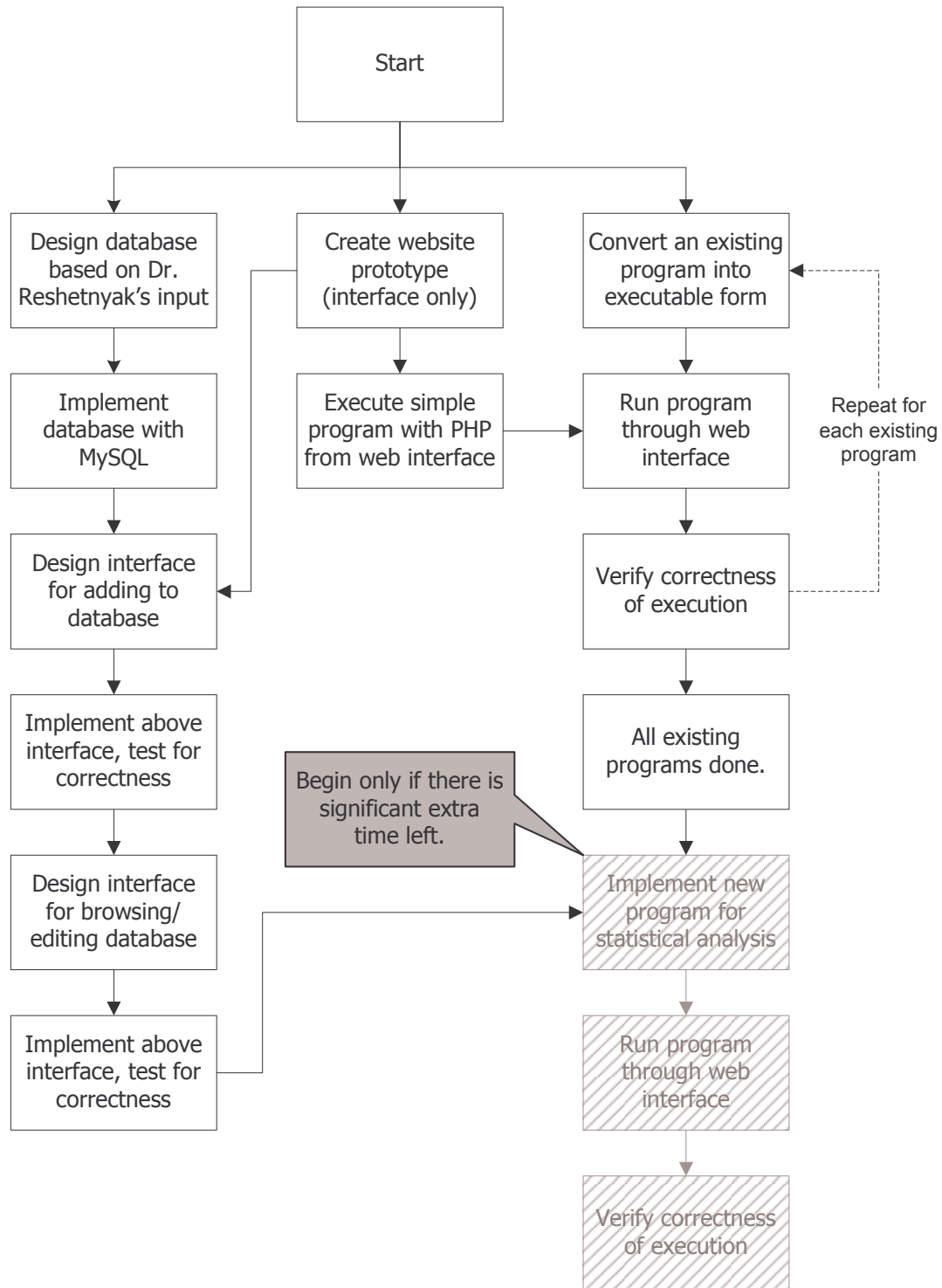
### Administrator (Yana)

**Revision date: May 18, 2005**



### 3.5 Implementation Plan

The following is our overall implementation plan for the components listed in the previous sections:



## 4. Results

### *4.1 Accomplishments*

We have implemented a web interface Dr. Reshetnyak's unique fluorescence database and some of her analysis tools.

The fluorescence database is intended to be an online reference resource for protein researchers, much like the PDB databank. It is a collection of verified experimental results that provide useful information about certain fluorescence and structural characteristics of proteins under various conditions. On a higher-level, this information is widely useful, including protein dynamics studies.

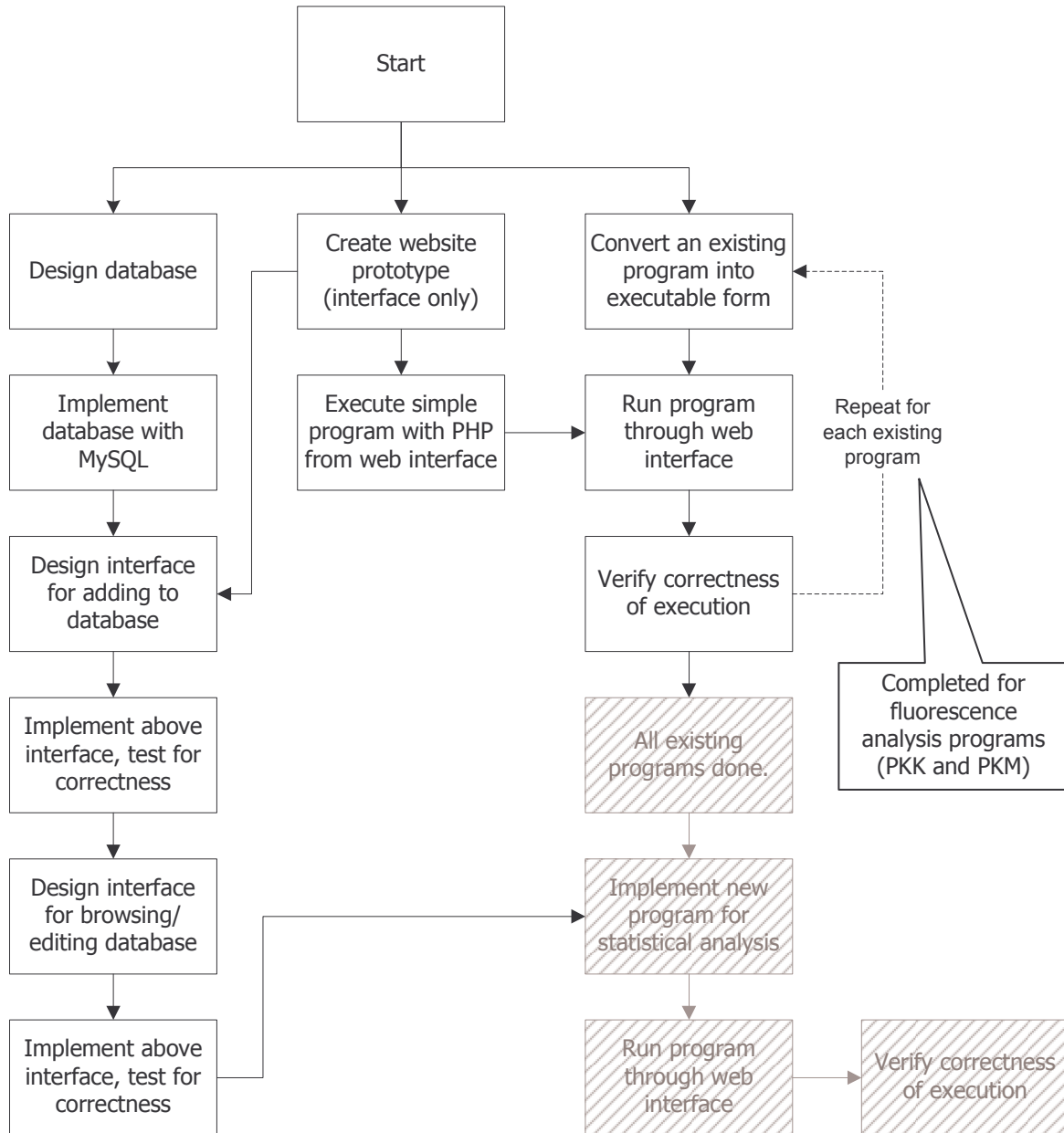
The fluorescence analysis tool can be used online and offline, depending on need. The online analysis is suitable for first-time users or for quick, short analyses. The tool can be downloaded for local use for more computationally demanding analyses. This tool, along with the structural and correlation analysis tool (unimplemented) allow researchers to obtain the same results as are in the database. Researchers may want to verify their own experimental results with the database, or analyze new experiments that are not yet in the database.

Our contribution has been to implement an online version of the database and providing a web interface for the analysis tools. This contribution is a first step in making the fruits of Dr. Reshetnyak's research more accessible to the rest of the world.

## 4.2 Implementation Status

## Implementation Plan

Referring back to our overall implementation plan, the follow represents what we have completed. Shaded items were not implemented.



## User Functionality

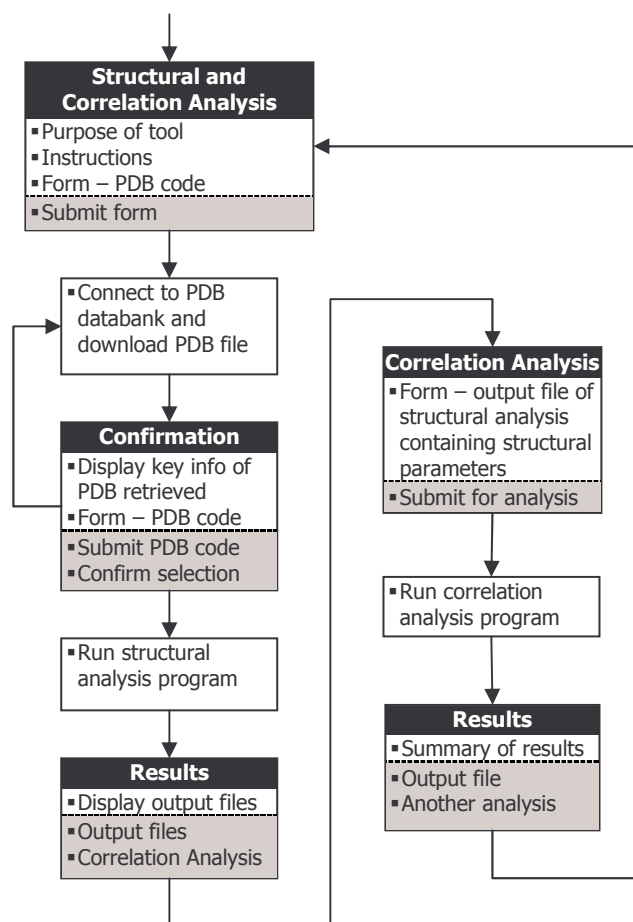
All user functionality specified in Section 3.2 has been implemented, except “Use Structural and Correlation Analysis Tool.”

## Database

The database has been implemented as specified in our design (in Section 3.3).

## Web Interface

We have implemented all parts of the web interface as specified by the website interaction diagrams (Section 3.4), except for the Structural and Correlation Analysis tool.



## 5. Project History

### 5.1 People

The following are the key people involved with this project. It consists of two professors and five graduate students.

**Dr. Yana Reshetnyak**, reshetnyak@mail.uri.edu, Department of Physics

The goal of the team is to help Dr. Reshetnyak's research. She is a biophysics researcher and has developed original techniques for using protein fluorescence to shed light on protein structure. She will provide the team with background and other relevant information for the project.

**Dr. Joan Peckham**, joan@cs.uri.edu, Department of Computer Science

Dr. Peckham is one of the professors for this bioinformatics class. She will provide general guidance for the project.

The student team is composed the following graduate students with various skills:

- **Nidhi Bansal**, nidybansal@gmail.com  
Web development, databases, programming
- **Neelima Guduru**, neelima\_guduru@yahoo.com  
Programming, web development, databases
- **Stephen Jaegle**, jaegles@cs.uri.edu  
Project management, programming, web development, graphic design, databases
- **Rajiv Menon**, rajiv.menon@gmail.com  
Web development, programming, database design
- **Chi Shen**, shenc@cs.uri.edu  
Project management, information systems, web development, programming, interface design, graphic design, database design

## 5.2 Task Assignments

*Numbered items represent tasks assigned; checked items represent progress or a finished task.*

Date	Chi Shen	Nidhi Bansal	Neelima Guduru	Rajiv Menon	Stephen Jaegle	Yana Reshetnyak
<b>Feb02</b>	(Taiwan trip)	✓ Revised and reformatted proposal	✓ Revised and reformatted proposal	✓ Revised and reformatted proposal	✓ Revised and reformatted proposal	
<b>Feb09</b>	(Taiwan trip)	✓ Finish proposal	✓ Finish proposal	✓ Finish proposal	✓ Finish proposal	
<b>Feb17</b>	1.Set up technical working environment	1.Learn PHP/MySQL basics	1.Learn PHP/MySQL basics			
<b>Feb23</b>	✓ Team website up, Apache/PHP/MySQL up, FTP accounts up.					
<b>Mar02</b>	1.Implement C++ program to parse PKK input file, document related variables	1.Determine database requirements and begin design	1.Web information architecture diagram	1.Work on PKK variable mapping document	1.Implement PHP external execution proof-of-concept	1.Email team something before next meeting
<b>Mar09</b>	✓ Finished program and doc 1.Revise proposal 2.Make implementation plan	✓ Met with Yana re: database 1.Continue with database, produce an ER diagram	✓ Drafted diagram 1.Revise and complete diagram	✓ Drafted variable document 1.Run PKK using VB	✓ Found PHP command ✓ Drafted initial database design 1.Study analysis programs and run using free BASIC	
<b>Mar11</b>	✓ Revised proposal background 1.Finish final proposal 2.Ask about change in scope 3.Start HTML pages based on Neelima's outline	✓ Revised proposal background	✓ Revised proposal background	✓ Successfully ran PKK, had some trouble with input file 1.Compile PKK for Win32	✓ Implementation complete 1.Find PHP graphing library and graph a PKK graph file	
<b>Mar16</b>	✓ Drafted main website pages ✓ Finished final proposal 1.Implement user registration 2.Implement FCAT	✓ Finished database design		✓ Ran PKK w/o problems		
<b>Mar23</b>	✓ Improved website, new FCAT pages, file upload, execute PKK online 1.Finish interaction diagram (from Neelima)	1.Implement MySQL database 2.Implement browse database pages	1.Implement MySQL database 2.Implement browse database pages	✓ Compiled PKK 1.Compile PKK in VB.NET		1.Revise PKK and send to team 2.Send and explain structural analysis program 3.Work with Steve on graph file 4.Send team conference details 5.Get a computer for use as PFAST server
<b>Mar29</b>	✓ Drafted user registration ✓ Drafted sign-in / sign-out ✓ Updated FCAT to use new PKK 1.Revise interaction diagram 2.Implement Admin Manage Database pages	✓ Implemented database ✓ Drafted browse database interface 1.Implement Admin Manage Users pages	✓ Implemented database ✓ Drafted browse database interface 1.Implement Admin Manage Users pages	✓ Compiled new PKK		✓ Revised PKK and sent out ✓ Sent explanation of structural programs and input files ✓ Sent conference details



<b>Apr06</b>	<ul style="list-style-type: none"> <li>✓ Revised interaction diagram</li> <li>✓ Made basic admin pages</li> <li>✓ Implemented save experiment to database</li> <li>✓ Updated browse database</li> </ul> 1.Update interaction diagram 2.Update database structure 3.Get Rajiv/Yana software			1.Copy CD and help Yana install software 2.Determine server specs for future PFAST server		✓ Computers are available
<b>Apr13</b>	<ul style="list-style-type: none"> <li>✓ Updated database structure</li> <li>✓ Updated interaction diagram</li> <li>✓ Got Rajiv/Yana software</li> </ul> 1.Revise FCAT interface			<ul style="list-style-type: none"> <li>✓ Helped Yana install software</li> <li>✓ Picked out server for Yana</li> </ul> 1.Compile new PKK 2.Work on biophotonics presentation		<ul style="list-style-type: none"> <li>✓ Handed out revisions to FCAT and PKK interface</li> <li>✓ Handed out draft of conference slide</li> </ul> 1.Revise PKM
<b>Apr20</b>	<ul style="list-style-type: none"> <li>✓ Revised FCAT interface</li> <li>✓ First draft of final report</li> </ul>	<ul style="list-style-type: none"> <li>✓ Draft transparency</li> <li>✓ Implemented "Delete User"</li> </ul>		<ul style="list-style-type: none"> <li>✓ Compiled new PKK</li> </ul> 1.Add content to final report	<ul style="list-style-type: none"> <li>✓ PHREQ graph works</li> <li>✓ Draft transparency</li> </ul> 1.Keep working on graphs	<ul style="list-style-type: none"> <li>✓ Emailed out new input files</li> <li>✓ Emailed out new PKK, PKM</li> <li>✓ Emailed out figures for transparency</li> </ul>
<b>Apr27</b>	<ul style="list-style-type: none"> <li>✓ Draft presentation text</li> <li>✓ Finished browse database</li> </ul> 1.Update interaction diagram 2.Add PKM to FCAT pages	<ul style="list-style-type: none"> <li>✓ Draft presentation text</li> </ul> 1.Implement "Manage Database"	<ul style="list-style-type: none"> <li>✓ Implemented "Edit User"</li> </ul> 1.Implement "Manage Database"	<ul style="list-style-type: none"> <li>✓ Draft presentation text</li> <li>✓ Compiled new PKK, PKM</li> <li>✓ Added content to final report</li> </ul>	<ul style="list-style-type: none"> <li>✓ Revised transparency layout, added graphics, revised text</li> <li>✓ Updated transparency</li> </ul>	<ul style="list-style-type: none"> <li>✓ Draft presentation text</li> <li>✓ More figures for transparency</li> </ul>
<b>Apr28</b>	<b>BIOPHOTONICS CONFERENCE – NIDHI WINS THIRD PLACE!</b>					
<b>May04</b>	<ul style="list-style-type: none"> <li>✓ Updated FCAT pages to handle PKM/PKK choice</li> </ul> 1.Write "Future Work" section of final report. 2.Fix database pages to properly display slashed data 3.Add web error-checking code for Rajiv's new PKK/PKM 4.Add "FCAT in progress" page	<ul style="list-style-type: none"> <li>✓ Finished "Manage database"</li> </ul> 1.Polish all administrative pages (breadcrumb navigation, stripslashes, table formatting, etc.) 2.Prepare to give 2-min presentation again for class next week	<ul style="list-style-type: none"> <li>✓ Finish "Manage database"</li> </ul> 1.Plan and outline presentation 2.Revise "Edit User"	1.Document database design for final report.	<ul style="list-style-type: none"> <li>✓ More graphs work</li> </ul> 1.Keep working on graphs 2.Make notes about graphs for final report "Problems Encountered" section	1.Check FCAT and database pages
<b>May10</b>	<b>CLASSES END</b>					
<b>May11</b>	<ul style="list-style-type: none"> <li>✓ Wrote "Future Work" section</li> <li>✓ Added web error-checking for new PKK/PKM</li> <li>✓ "FCAT in progress" page is not feasible</li> </ul>	<ul style="list-style-type: none"> <li>✓ Fixed bugs in Administration web pages</li> </ul>	1.Finish outline for presentation			✓ Revised interaction diagram
<b>May17</b>	<ul style="list-style-type: none"> <li>✓ Revised report (Executive Summary, Database Design, Future Work, User Functions, Accomplishments, Implementation Status)</li> <li>✓ Revised interaction diagram</li> </ul>	<ul style="list-style-type: none"> <li>✓ Fixed bugs in Administration web pages</li> <li>✓ Populated database with realistic data</li> </ul> Revised report (Problems, Lessons Learned, Defect Report)	<ul style="list-style-type: none"> <li>✓ Prepared presentation slides</li> </ul>	<ul style="list-style-type: none"> <li>✓ Prepared presentation slides</li> </ul>	<ul style="list-style-type: none"> <li>✓ Revised report (Problems, Lessons Learned)</li> </ul>	
<b>May19</b>	<b>CLASS FINAL PRESENTATION</b>					

## ***5.3 Problems Encountered***

### **Lack of project domain experience**

All team members are computer science graduate students with strong technical backgrounds, but little or no background in Dr. Reshetnyak's area of research. We took a long time to understand well the significance of the project and data to be stored in the database.

### **Decision between rewriting or using existing code**

In the beginning, the team got deep into understanding the algorithms and the existing programs with the intention of rewriting them. We realized after a time that we had lost focus of the main goal of the project: to improve accessibility of Dr. Reshetnyak's analysis programs. Thus, we decided to reprioritize and save this work for later.

### **PHP Graphing**

Because the implementation of PFAST is online, displaying its graphs calls for a solution that will work well in an online server environment. A solution for plotting that runs in PHP is thus highly desirable because of platform independence and portability. PHPlot uses the GD library which is part of most PHP configurations. Notably, the GD library is not included in the windows implementation of PHP prior to version 5, which proved as source of some initial implementation frustration, but the library nevertheless generally widely available. PHPlot can produce a variety of plots, including lines and points required for the current implementation, as well as offering other types that may be required in the future, and allows the combination of different types of plots in a single graphic, though not quite in the way originally envisioned - if the plots have differing numbers of points, considerable extra work is required. Nevertheless, PHPlot is able to generate dynamic plots of multiple types quickly. Also, PHPlot is open source, and licensed under the GNU GPL and the PHP License.

## ***5.4 Lessons Learned***

Implementing the plots for this project in PHPlot proved far more difficult than anticipated. PHPlot's description of capabilities states that it can produce plots that combine multiple styles, but the actual capability falls short of combining plots in the precise ways required by PFAST. A considerable number of workarounds will be required to implement the combined plots of PFAST using PHPlot. Also, gaps in the documentation and made for extra implementation work. In one instance, for example, the code shown for generating a sample plot did reproduce the sample plot properly when loaded and executed. Although PHPlot was in the last analysis chosen over some other PHP plotting packages partly because of the availability of drawing libraries in PHP on the Windows platform, a clear lesson is that a more in-depth testing process beyond the examining the description and sample plots might have revealed its limitations earlier in the development process.

## **6. Future Work**

There are many areas for future work.

### **Rewrite Analysis Tools**

The analysis tools should be rewritten in a cross-platform and compiled language. The tools are currently written in Visual Basic, which is interpreted and dependent on the Windows platform. The motivation for using a compiled language is to improve the performance (execution time) of the tools. The motivation for using a cross-platform is to remove the dependence on Windows and Visual Basic and also to allow for wider distribution and use of the tools.

### **Implement Structural Analysis Tool**

The structural analysis tool has not been implemented. The basic steps for implementing the tool are as follows:

1. Display a form for inputting a PDB code.
2. The system connects to the PDB databank to retrieve appropriate PDB file.
3. The analysis program processes the PDB and output structural parameters.
4. The administrator has the option of saving the parameters to the PFAST database.

### **Accommodate Heavy Use of FCAT**

If there is heavy use of the FCAT tool for extended analyses, the system will need to be revised to handle various concurrency issues. First, it may be necessary for the system to limit the number of FCAT analyses that can be running simultaneously. If the limit has been reached, the system could offer the user to be placed in a queue. Also, the system will need to offer the user a way to terminate an analysis without waiting for it to finish.

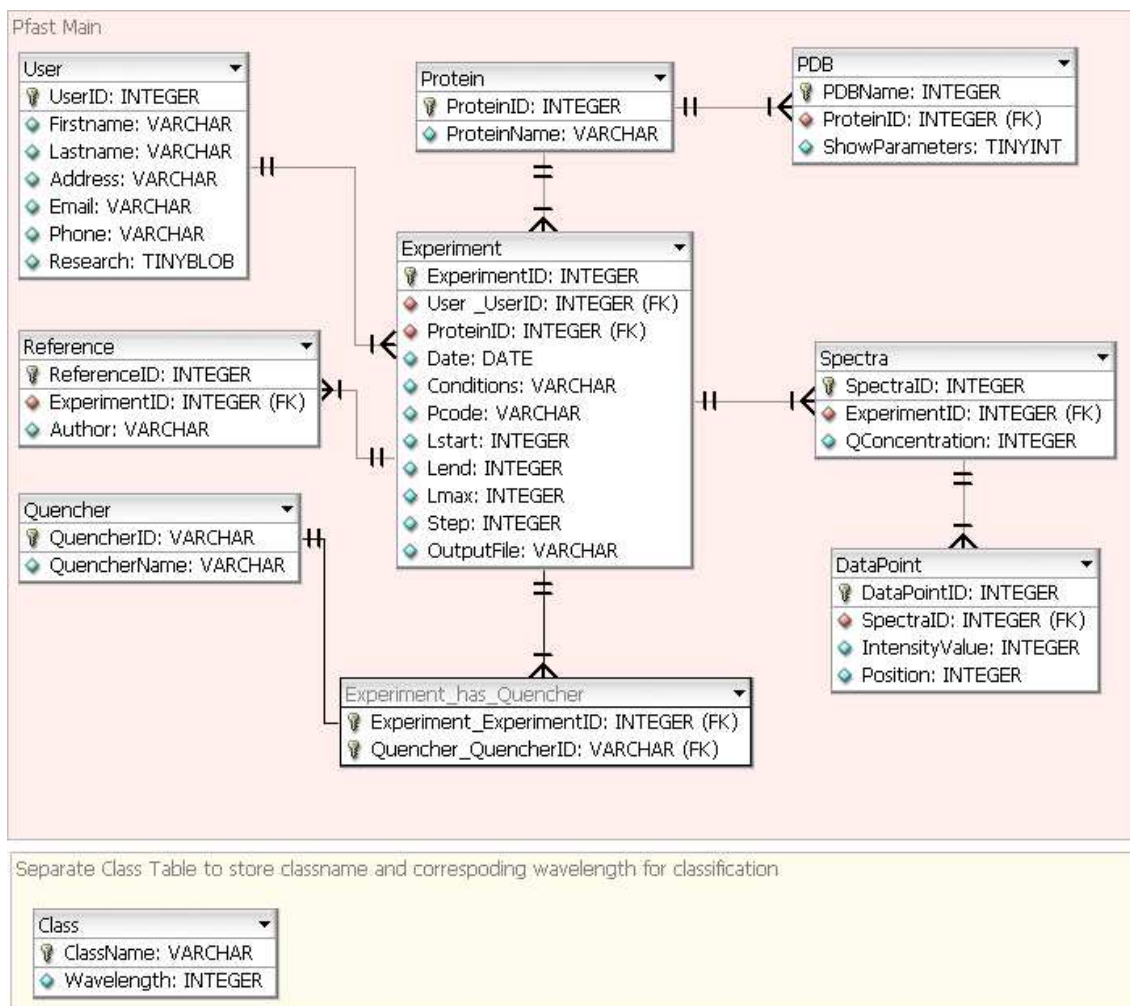
### **Migrate to Dedicated Server**

The system needs to eventually be moved to a dedicated server. The basic steps are:

1. Install and configure software (Apache - web server, PHP - web scripting, MySQL - database package) on the server
2. Copy website files from current computer to the server.
3. Export the database schema and data from current computer and import it into the instance of MySQL on the server.
4. Configure domain name to point to the IP address of the new server.

## Normalize Database

When the database has matured to a point where the underlying structure is no longer changing, it may be worthwhile to normalize the database for many reasons. It is beyond the scope of this report to discuss the benefits of database normalization, and the reader should find other references regarding this topic. However, we give a database schema that could be the foundation for future normalization efforts.



## 7. References

- [1] Edward A. Burstein, Sergei M. Abornev, and Yana K. Reshetnyak, **Decomposition of Protein Tryptophan Fluorescence Spectra into Log-Normal Components. I. Decomposition Algorithms**, Biophys. J. 2001 81: 1699-1709, <http://www.biophysj.org/>
- [2] Yana K. Reshetnyak, Yuly Koshevnik, and Edward A. Burstein, **Decomposition of Protein Tryptophan Fluorescence Spectra into Log-Normal Components. III. Correlation between Fluorescence and Microenvironment Parameters of Individual Tryptophan Residues**, Biophys. J. 2001 81: 1735-1758, <http://www.biophysj.org/>