

Florida International University
School of Computing and Information Sciences

CIS 4911 - Senior Capstone Project
Software Engineering Focus

Final Deliverable

GenomePro 3.0

Team Members

Daniel Gonzalez (3760013)

Roberto Arciniegas

Product Owner: Michael Robinson

Instructor: Masoud Sadjadi

Copyright Notice

Copyright © 2015 by Michael Robinson. All Rights Reserved.

The materials available through this document and the GenomePro website owned or provided by the here signed authors are the property of GenomePro or its licensors, and are protected by copyright, trademark and other intellectual property laws. No part of this documentation may be reproduced, distributed, or transmitted in any form or by any means, including photocopying, recording, or other electronic or mechanical methods, without the prior written permission from the authors, except in the case of brief quotations embodied in critical reviews and certain other noncommercial uses permitted by copyright law. For permission requests, write to Michael Robinson, addressed “Attention: Permissions Coordinator,” at the address: michael.robinson@cs.fiu.edu .

Trademark Notice

GenomePro, the logo, and other trademarks are trademarks of Michael Robinson and may not be used without permission.

Abstract

The Genome Project was launched about a decade ago. After its launch, the need for new considerations on genetic aids to human diseases and the development of coherent strategies for decreasing disease phenotypes altogether was the starting point for gene hunters to start mining wonders from the human genome. These professionals needed more powerful tools and more ambitious strategies to aid their work; hence, the application of computational methods to genomic data science became an ever-growing and ever-more-exciting field developing both the biological and medical academic and professional fields. With these thoughts in mind, GenomePro conveys a well-structured programming framework designed to facilitate genomic analysis using robust and efficient tools for next-generation genome sequencing. The tools available through GenomePro provide a rich set of data-access patterns that encompass the majority of genomic analysis' requirements, thus offering users the power to process dauntingly large genomic data files. GenomePro also generates resulting data accurately, detailing the differences and similarities between genome files while also maintaining statistics about the data analyzed.

Table of Contents

Introduction.....	6
Current System.....	7
Purpose of New System.....	8
User Stories.....	8
Project Plan.....	12
Hardware and Software Resources.....	13
Hardware:.....	13
Software:.....	13
Sprint Plan.....	14
Sprint 1.....	14
Sprint 2.....	15
Sprint 3.....	16
Sprint 4.....	17
Sprint 5.....	18
Sprint 6.....	19
Sprint 7.....	20
System Design.....	21
Architectural Patterns.....	21
Design Patterns.....	21
System and Subsystem Decomposition.....	22
Class Descriptions.....	23
Deployment Diagram.....	26
System Validation.....	27
Glossary.....	40
Appendix.....	41
Appendix A - UML Diagrams.....	41
Static UML Diagrams.....	41
Dynamic UML (Sequence) Diagrams.....	49
Appendix B - User Interface Design.....	56
Appendix C - Sprint Planning Reports.....	65
Appendix D - Sprint Review Reports.....	68

Sprint 1 Review Report.....	68
Sprint 2 Review Report.....	68
Sprint 3 Review Report.....	69
Sprint 4 Review Report.....	70
Sprint 5 Review Report.....	70
Sprint 6 Review Report.....	71
Appendix E - Sprint Retrospective Reports.....	72
Sprint 1 Retrospective.....	72
Sprint 2 Retrospective.....	73
Sprint 3 Retrospective.....	74
Sprint 4 Retrospective.....	75
Sprint 5 Retrospective.....	76
Sprint 6 Retrospective.....	77

INTRODUCTION

This section aims to identify and describe both the structure of the new system, in comparison with its previous version (2.0) and how it will be created and designed. This is version 3.0. We envision the system as a data processing framework designed to assist the work of biological and medical researchers. As such, the user interface should be simple, straightforward, and intuitive. The data processed by the system should always be validated and flagged as DNA, RNA, Protein, or Invalid data. Thus, GenomePro conveys a well-structured programming framework designed to facilitate genomic analysis using robust and efficient tools for next-generation genome sequencing. The tools available through GenomePro provide a rich set of data-access patterns that encompass the majority of genomic analysis' requirements, thus offering users the power to process dauntingly large genomic data files. The system is conceived at no cost to the user. This section also covers the hardware and software system requirements as well the work breakdown performed during the 8 sprint phases of development.

CURRENT SYSTEM

A Decade ago, the history of biology was forever transformed after an audacious decision to launch a research program that would depict, in critical detail, the complete set of genetic instructions of the human being – *The Genome Project*. The result of this development would include a new understanding of genetic contributions to human diseases and the development of coherent strategies for decreasing disease phenotypes altogether.

Not surprisingly, the disease researchers wanted to be able to analyze data as soon as possible and in the least time necessary. However, this came with some complications, as the not so small portion of discovered human genes scatters throughout the entire human genome like stars in the universe; yet, with genomic material representing light-years of noncoding DNA – *The Dark Matter*.

"Gene Hunters" who mine wonders from the human genome need more powerful tools and more ambitious strategies. Hence, the application of computational methods to genomic data science is an ever growing and exciting development, in both biological and medical fields of study.

GenomePro helps professionals of these fields by offering unheard of specific tools and methods for sequence processing with a robust structured analysis method that leads to the predictions of undiscovered genes. This structure is thus useful for managing large volumes of data, making its

representation and description more compact for better analysis and delivering a high degree of accuracy over long stretches of DNA sequence.

PURPOSE OF NEW SYSTEM

The purpose of the new system is to create a web application that will be able to automate and accelerate the task of processing staggeringly large genomic data files. The new system will offer users the power to efficiently process genomic data files and generate resultant documents containing accurate information about the differences, similarities, and various statistics between/concerning the associated input data files. In addition, a link to the resultant document will be emailed to users once the system is done processing the input data. Moreover, the new system will allow the users to create and manage their account in which they can keep track of a history of jobs submitted by them and results processed by the system. The new system will also eliminate the need to manually perform these data calculations done by users, which can considerably reduce the amount of errors the users can have while doing it manually. Additionally, the new system will provide to the users the speed, performance and accuracy to make their research advance in a faster pace, while allowing users to parallelize job execution using GenomePro tools.

USER STORIES

This section presents a list of all user stories collected during the requirement analysis for the new system. They provide a common thread that weaves through the variety of artifacts necessary to support the work in development such as project priorities and estimates, flow diagrams, touchpoint designs, user interface designs, to name a few. These user stories allowed the project team flexibility in how to respond to particular hurdles encountered during development phases; thus, permitting innovative ways for creating system value while still achieving precision on the overall project vision and priorities. These user stories are functional and prioritized as part of the project work plan to help ensure that implementation tasks were targeted at high-value activities.

User Story # 280 – Prerequisites/Acquaintance with Pre-existing Software Solution

Priority: Required

Description:

- ✓ As a developer, I want to be prepared with sufficient background knowledge to tackle the problems that await for releasing the next version of this software package as well as having intimate knowledge of the workings and architecture of the software.

User Story # 281 – Prerequisites/Acquaintance with existing software

Priority: Required

Description:

- ✓ As a developer, I want to prepare with sufficient knowledge to continue the development of the GenomePro software package.

User Story # 282 – Redesign Tools Page (front end)

Priority: Required

Description:

- ✓ As a user, I want a more comprehensible and intuitive layout for the tools page wherein I can select the tool, file(s), and arguments from a series of horizontally-arranged drop-down menus, text boxes, and other forms of selection.

User Story # 283 – FTP file manager plug in

Priority: Required

Description:

- ✓ As a user, I want to be able to create, read, update and delete files from the genome pro server.

User Story # 285 – Redesign Tools Page (functionality)

Priority: Required

Description:

- ✓ As a user, I want to be able to submit multiple jobs at once via a more intuitively arranged tools page.

User Story # 286 – Upload Files Tool

Priority: Should

Description:

- ✓ As a user, I want to be able to upload files to my account without having an obtrusive web element which detracts from the stream-lined approach of the tools page.

User Story # 287 – Create a log to track system usage

Priority: Required

Description:

- ✓ As an administrator, I want to have a log to keep track of users logins.

User Story # 288 – Profile page redesign

Priority: Required

Description:

- ✓ As a user, I want my profile page to look more like a traditional, social-media-inspired, profile page, including a description box and information indicating how long I have been a user.

User Story # 289 – Move pending jobs to the profile page

Priority: Required

Description:

- ✓ As a user, I want to be able to see what jobs I have pending, as well as my job history, on my profile page as opposed to having to navigate to the tools page to see this information.

User Story # 290 – User log-in statistics

Priority: Must

Description:

- ✓ As a user, I want to be able to see usage statistics about how many people use GenomePro.

User Story # 291 – Upload File Integration with Multiple Job Submission

Priority: Could

Description:

- ✓ As a user, I want to be able to upload files while, at the same time, submitting multiple jobs.

User Story # 292 – Sign-in with email

Priority: Should

Description:

- ✓ As a user, I want to be able to log into GenomePro using my email address instead of my username if I wish to do so.

User Story # 293 – Correct file names for jobs

Priority: Must

Description:

- ✓ As a user, I want the file names produced by successful jobs to accurately reflect the time at which the job was finished and the file was created.

User Story # 294 – Create a forum to share information with other users

Priority: Required

Description:

- ✓ As a user, I want to share information with other users, read the information that other users have to share and comment on it.

User Story # 295 – User usage statistics

Priority: Required

Description:

- ✓ As a system administrator, I want to see the the usage of the system.

User Story # 296 – Preparing for the Final Showcase

Priority: Required

Description:

- ✓ As a developer, I want to have all of my documentation prepared and all of the requirements for this course met so that I am fully prepared for the Final Showcase on Friday, May 6th, 2016.

User Story # 297 – Final release preparation

Priority: Required

Description:

- ✓ As a developer, I want to have all of my documentation prepared and all of the requirements for this course met so that I am fully prepared for the Final Showcase on Friday, May 6th, 2016.

PROJECT PLAN

This section aims to document the managerial aspects of the project which are required to execute it successfully amid all possible known constraints. The organizational breakdown is outlined here

together with the sprints that illustrate all user stories formerly implemented. The focus has been geared towards the development of a friendlier user interface and a more robust tools management interface.

Hardware and Software Resources

Hardware:

- ***PC - Windows OS (7+), Mac (OS X or higher), and Linux (Ubuntu)***

All work was done directly on the Linux server hosting GenomePro 3.0. Mac systems were used primarily for website media and testing.

Software:

- ***Mingle*** (Software development management tool, required by instructor)
- ***GitHub*** (Collaborative platform, required by instructor)
- ***Google Drive*** (Cloud storage service tool, required by instructor)
- ***PostgreSQL*** (Open source Object-Relational database system, required by product owner)
- ***PHP*** (Open source general-purpose scripting language, required by product owner)
- ***C*** (Programming language, required by product owner)
- ***HTML*** (Standard structural web pages design language)
- ***CSS*** (Style sheets necessary for structuring layouts of all web pages)
- ***JavaScript*** (Common client side scripting language to handle user interaction with web pages)
- ***jQuery*** (JS Library for traversal, manipulation, animation and event handling on HTML documents)
- ***StarUML*** (Open source Software-Modeling tool use for UML diagrams)
- ***PHPPGADMIN*** (UI for PostgreSQL)
- ***Skype*** (Collaborative source of communication in voice and video format)
- ***Microsoft Expression Web 4.0*** (Used for Html Site design format)
- ***Adobe Dreamweaver CS3*** (Used for Html Site design format and Media enhancement)
- ***Adobe Premiere Pro CS3*** (Used for Video Making and Editing)

Sprint Plan

During each sprint, visual artifacts were created, thus driving the team towards a spirit of work completion with the opportunity to show off new work at each sprint demo. Valid incremental feedback from stakeholders during these demos, allowed for improvement of team productivity and confidence in the product development process as a whole. Below is a brief description of each sprint, based on each implemented user story.

Sprint 1

(01/18/2016 – 01/29/2016)

User Story # 280 – Prerequisites/Acquaintance with Pre-existing Software Solution

Owner: Daniel Gonzalez

Tasks:

1. Work Through Tutorial Videos

Acceptance Criteria:

- a. Finish all videos up to and including the "TalkingSpace Forum" on StackSkills
- b. Feel comfortable presenting the software solution in full depth and detail to an audience
- c. Obtain proficiency with the tools and languages required for development on the software product

User Story # 281 – Prerequisites/Acquaintance with existing software

Owner: Roberto Arciniegas

Tasks:

1. #301 Build Test System

Acceptance Criteria:

- f. Finish the StackSkills tutorial on PHP
- g. Feel comfortable presenting the software solution in full depth and detail to the audience
- h. Obtain proficiency with the tools and languages required for development of the product

Sprint 2

(2/1/2016 – 02/12/2016)

User Story # 282 – Redesign Tools Page (front end)

Owner: Daniel Gonzalez

Tasks:

1. **#303** Create Drop-Down Menus
2. **#304** Create File Selector & Arguments
3. **#305** Create Buttons

Acceptance Criteria:

- a. Intuitive and visually ergonomic
- b. Must have a drop down menu to select the tool to use for the selected job
- c. Must have a way to select the files to apply the tool to
- d. Must have a text box to enter any additional arguments for the (appropriately selected) tool
- e. Must have a button for submitting the job to the server
- f. Must have the preceding four elements arranged sequentially and horizontally

User Story # 283 – FTP file manager plug in

Owner: Roberto Arciniegas

Acceptance Criteria:

- a. GUI simple in design and intuitive
- b. The FTP interface should open in a new tab
- c. Should be accessible from any page
- d. Should be able to upload, download, delete, and rename files

Sprint 3

(02/15/2016 – 02/26/2016)

User Story # 285 – Redesign Tools Page (functionality)

Owner: Daniel Gonzalez

Tasks:

1. **#306** Add Row Button
2. **#307** FASTA/Probes Arguments

Acceptance Criteria:

- a. "+" button which adds a new row of inputs for submitting an additional job within the same form
- b. Arguments text field for "Extract FASTA" appears when the appropriate tool is selected
- c. Arguments text field for "Probes" appears when the appropriate tool is selected

User Story # 287 – Create a log to track system usage

Owner: Roberto Arciniegas

Acceptance Criteria:

- a. The system will track the user id
- b. The system will track the time stamp of the login event
- c. The system produces reports of the login events activities

Sprint 4

(2/29/2016 – 3/11/2016)

User Story # 286 – Upload Files Tool

Owner: Daniel Gonzalez

Tasks:

1. **#308** Implement Upload Tool

Acceptance Criteria:

- a. A new tool is implemented to the tools drop-down menu called "Upload files"
- b. When that tool is selected, a file browser is displayed for the user to select his/her files with
- c. Upon submission, the software system will correctly update the database with the user's new files

User Story # 292 – Sign In With Email

Owner: Daniel Gonzalez

Tasks:

1. **#310** Modify Queries

Acceptance Criteria:

- a. Users can now log into GenomePro using their email address
- b. Users can still log into GenomePro using their username as before

User Story # 290 – User log-in statistics

Owner: Roberto Arciniegas

Acceptance Criteria:

- a. GenomePro now displays how many users log in per unit time (for ex: per month)
- b. Information on how often each specific user logs in is now visible on their profile page

Sprint 5

(03/21/2016 – 04/01/2016)

User Story # 288 – Profile Page Redesign

Owner: Daniel Gonzalez

Tasks:

1. #311 Add Description Box
2. #312 Add Registration Badge

Acceptance Criteria:

- a. Profile pages now have description boxes capable of being edited by users
- b. Profile pages now have badges indicating how long users have been registered with GenomePro

User Story # 289 – Move pending jobs to the profile page

Owner: Daniel Gonzalez

Tasks:

1. #313 Move Jobs Table

Acceptance Criteria:

- a. The Job History module is now visible in the profile page
- b. The Job History module is no longer visible on the tools page

User Story # 293 – Correct File Names For Jobs

Owner: Roberto Arciniegas

Acceptance Criteria:

- a. File names now accurately reflect the timestamp for when the job was finished processing by
GenomePro

Sprint 6

(04/04/2016 – 04/15/2016)

User Story # 291 – Upload File Integration with Multiple Job Submission

Owner: Daniel Gonzalez

Tasks:

1. #314 Integrate File Upload with M.J.S.

Acceptance Criteria:

- a. Users are now able to submit files along with the submission of multiple jobs via the tools page

User Story # 294 – Create a forum to share information with other users

Owner: Roberto Arciniegas

Acceptance Criteria:

- a. All users will be able to read the forum
- b. Register users will be able to write in the forum
- c. Register users will be able to reply to postings from other users

User Story # 295 – User usage statistics

Owner: Roberto Arciniegas

Acceptance Criteria:

- a. The usage page will display the successful login events
- b. The usage page will display the job submissions
- c. The page will show the information sorted chronologically or by user

Sprint 7

(04/18/2016 – 04/29/2016)

User Story # 296 – Preparing for the Final Showcase

Owner: Daniel Gonzalez

Tasks:

1. **#315** Create Poster
2. **#316** Fix Mingle
3. **#317** Prepare Final Document

Acceptance Criteria:

- a. All Feature Documents are uploaded and consistent with Mingle
- b. Mingle is reflective of my [Daniel Gonzalez 3760013] workflow for this [GenomePro 3.0] project
- c. Poster is prepared
- d. All Meeting Minutes are uploaded

User Story # 297 – Final release preparation

Owner: Roberto Arciniegas

SYSTEM DESIGN

This section provides a detailed description of the design architecture and patterns used in the GenomePro application. It refers to class diagrams in Appendix A which contains a depiction of all sub-systems implemented, as well as the sequence diagrams for each use case implemented describing the object interactions. Finally this section provides a brief statement of purpose of each class.

Architectural Patterns

As a website application, GenomePro is bound to many pre-existing conventions and technical requirements that come with developing on such a platform. Our chosen architecture is not innovative or new, but purely another adaptation of common patterns used by many of the million web applications that already exists in place. The advantage to this, is that our architecture incurs low risk in implementing, since it has already been tried and tested many times over.

The system implementation was done using the **Model – View – Controller (MVC)** as the primary architectural pattern for the GenomePro system. MVC is the basic architecture for most web applications, mainly because this platform has been designed in such a way that it forces web applications into such a setup. Nevertheless, this architecture is still appropriate for our system and we are glad to work with it, because it emphasizes modularity and scalability allowing the separation of the application's data, logic, and user interface.

Design Patterns

Under the MVC architecture, the system is split into 3 packages: the UI package, the Controller package, and the Database package. These packages implement 3 design patterns, as seen in the class diagrams shown in Appendix A. The following is a brief explanation:

Singleton Pattern: GenomePro uses the Singleton Design Pattern by creating classes to manage access to all data the system will have access to and the history of all job results the user has submitted.

Façade Pattern: Our system uses the Facade Pattern in the Controller package as a way to make the system easier to understand, use and test. One of these classes is the `system_helper.php` which controls every page in the UI.

Command Pattern: We use the Command Pattern as a way to generalize all the different web pages by using a common class called `templates.php`. This class implements all pages in the system, e.g. History, Contact, Home and Tools.

System and Subsystem Decomposition

All three components: The Models being the data being manipulated, the Views comprising the user interface, and Controllers containing the business logic that manipulates Models and Views communicate with each other through clearly defined interfaces. This makes it simple to swap components, as long as new components use the same existing interfaces. For example, if we decided to switch from remote to local database in the future, we would only need switch out the “Model” component of our application. Also, if more tools are later aggregated we can simply create new templates for such tools.

The **Model Subsystem** is composed of the objects that provide the unique capabilities and information storage for the application; It contains all the rules for processing application data. In fact, this is the key sub-system that makes the application valuable. In GenomePro the classes for the Model

The **View Subsystem** grants information gathered from the Model providing a way for users to interact with this data.

The main purpose of the **Controller Subsystem** is to decouple the Model from the View. It also handles data formatting for presentation to the user.

MVC System Design for GenomePro

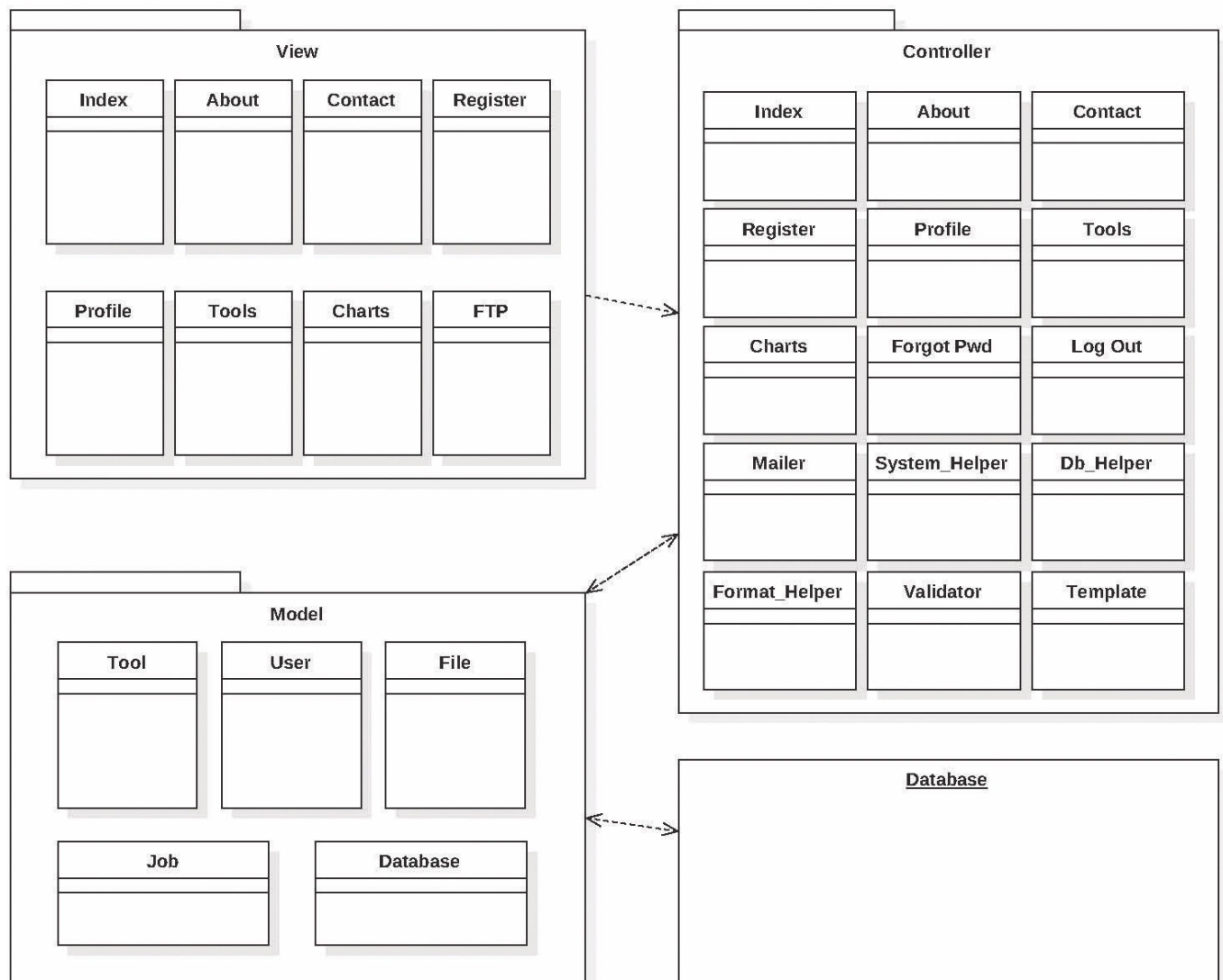


Figure 1 Depiction of GenomePro 3.0 System and Subsystem Decomposition

Class Descriptions

To better understand the Classes Diagrams in the GenomePro system as referred in the Appendix A of this document, please see below a brief description of each class contained inside the controller and model packages of the above Class Diagram.

Controllers:

- ***Index Controller:*** Controls the main page of the site. Handles 'login' use cases by collecting and storing user information. Also handles 'forget password'.
- ***About Controller:*** This controller only prints out HTML data without establishing any real communication with models.
- ***Tools Controller:*** This controller collects tools, jobs, and files of a logged in user. It passes this information to the view in which the user will use to submit a job. The controller also processes such jobs.
- ***Logout Controller:*** Doesn't communicate with any views or models, but it handles the logout case. Accessed by any page, this controller simply deletes PHP sessions.
- ***Register Controller:*** Handles user registration. Doesn't send anything to any views, but collects information from the Register page. It processes the 'register' and 'confirmation of email' use cases through models, mainly, the User model.
- ***Profile Controller:*** Collects user information from models, including name, jobs submitted, etc. Displays this information on the Profile view. Also handles certain use cases from the view, including 'changing password' and 'changing email'.
- ***Contact Controller:*** This controller doesn't connect to any particular model, but uses another controller – the System Helper controller, to send emails to the server. It collects information on what to write in the email by what is inputted by the user in the Contact page.
- ***System Helper Controller:*** This controller doesn't act on its own, but is used by almost all other controllers. It provides helpful features, including sending emails, uploading files to the server, uploading avatars, etc.
- ***Database Helper Controller:*** Currently unimplemented. Doesn't do anything as of now.
- ***Format Helper Controller:*** A controller used mainly by views. Its task is to format certain aspects of the website. For example, if the TIMESTAMP of a database must be converted to an easier to understand format, this controller handles such tasks.

Models:

- **Database:** Only model that connects to the database and any queries must go through here. Uses prepared statements for security, and is scalable to use with any database system.
- **File:** Model that handles queries with files. This model is used by any controllers that need information about a particular file, or needs to gather a series of rows from an arbitrary amount of users.
- **Job:** This model deals with the submission and retrieval of jobs. Used mainly by the Tools controller, this model handles queries to insert new jobs as a row in the Job table.
- **Tool:** A very simple model, which collects information about the kind of tools GenomePro currently offers. These are stored in the database with a name and a description, as well as a type, which signifies the kind of input the tool takes.
- **User:** The most complex model in the system. This model handles anything that is solely user related, including login in, logout, register, etc. As such, this model also relies on the use of queries to process information.
- **Template:** Although this model doesn't connect to the database, it creates a platform to allow controllers to connect with views. Using a template, a controller can assign variables, and print out the view with those variables 'baked in'.
- **Validator:** This is another model that does not connect to the database. It has the task of verifying input, including emails, passwords, etc. Used by a lot of controllers that want to validate if something is correct or not.

Deployment Diagram

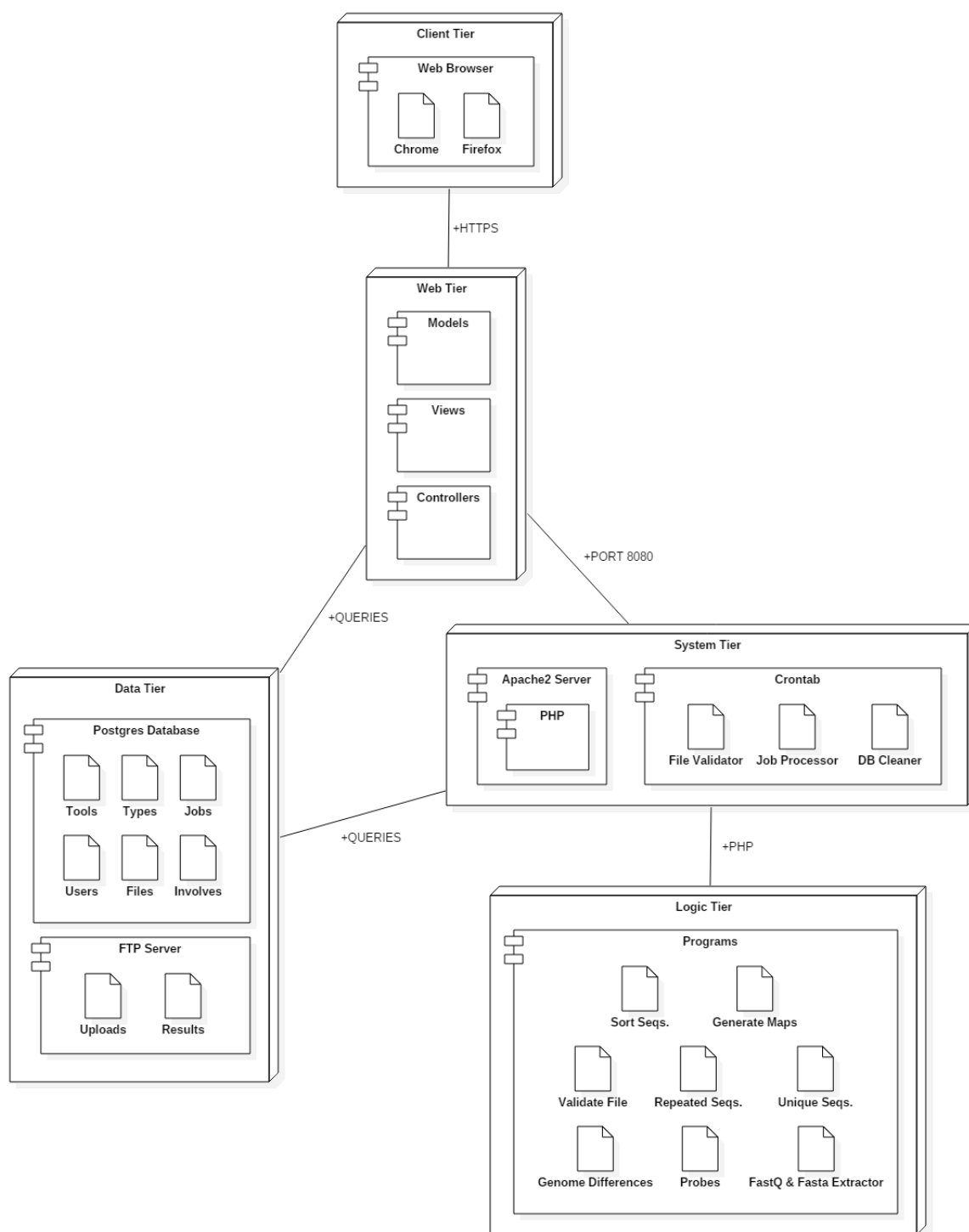


Figure 2 - Deployment Diagram for the GenomePro 3.0 Website as of Sprint 7

SYSTEM VALIDATION

This chapter is dedicated to the testing performed on the system in order to validate and verify the functions implemented in the GenomePro system against the specified requirements. System testing was done in order to validate the user interface, controllers, storage, and the system as a whole. Additionally, unit and integration testing were performed on each use case involving the tools page to assure that there were no unforeseen side-effects caused by redesigning elements of the Tools Page or modifying the job submission algorithm found on the Tools Page. There was also some unit and integration testing carried out on other use cases of merit.

Test case #: 1	Test case name: Submit Job (sunny)	User Story #: 282
Description:	Verify that users are able to submit jobs correctly if they follow the correct protocol.	

Pre-conditions:	1. User has successfully logged into the system. 2. User has navigated to the tools page.
------------------------	--

Steps	Action	Value
1.	User selects a tool from the drop-down menu.	Repeated Sequences
2.	User selects a file from the drop-down menu.	201602011319_fasta.txt.sorted
3.	User presses the "Submit" button.	N/A
Result: ✓ Passed		

Post-conditions:	The User is presented with a reloaded tools page and a message indicating that the job completed successfully, along with the location of their resultant file.

Test case #: 2	Test case name: Submit Job (rainy)	User Story #: 282
Description:	Verify that the system correctly detects errors in the submission process	

Pre-conditions:	1. User has successfully logged into the system. 2. User has navigated to the tools page.
------------------------	--

Steps	Action	Value
1.	User does not select a tool from the drop-down menu.	None
2.	User presses the "Submit" button.	N/A
Result: ✓ Passed		

Post-conditions:	The User is presented with a reloaded tools page and a message indicating that the system was unable to process the job request due to the user not having selected a file.

Test case #: 3	Test case name: Update profile (sunny)	User Story #: 288
Description:	Verify that users are able to update their profile if they enter correct information into the fields.	

Pre-conditions:	1. User has successfully logged into the system. 2. User has navigated to their profile page.
------------------------	--

Steps	Action	Value
1.	User enters a new email address.	musiqueestmaths@gmail.com
2.	User enters their password.	p4ssword
3.	User enters their password again.	p4ssword
4.	User enters a description.	"Hello :)"
5.	User clicks the "Submit" button.	N/A
Result: ✓ Passed		

Post-conditions:	The User is presented with a reloaded profile page with all changes effectuated as well as a message indicating that the profile has been successfully updated.

Test case #: 4	Test case name: Update profile (rainy)	User Story #: 288
Description:	Verify that users cannot enter malicious information or engage in XSS while updating their profile.	

Pre-conditions:	1. User has successfully logged into the system. 2. User has navigated to their profile page.
------------------------	--

Steps	Action	Value
1.	User enters a malicious email address.	; DROP TABLE users;
2.	User enters their password.	p4ssword
3.	User enters their password again.	p4ssword
4.	User enters a description.	"Hello :)"
5.	User clicks the "Submit" button.	N/A

Result: ✓ Passed

Post-conditions:	The User is presented with a reloaded profile page with an error message indicating they have entered an invalid email address. No changes are made to either the server or the profile page.

Test case #: 5	Test case name: Log In (sunny)	User Story #: 289
Description:	Verify that users can successfully log into GenomePro using their credentials.	

Pre-conditions:	1. User has successfully registered an account with GenomePro. 2. User has navigated to the GenomePro front page.
------------------------	--

Steps	Action	Value
1.	User enters their username.	dgonz293
2.	User enters their password.	p4ssword
3.	User clicks the "Login" button.	N/A

Result: ✓ Passed

Post-conditions:	The User is presented the "logged in" version of the index page, along with a message indicating that they have successfully logged into GenomePro.

Test case #: 6	Test case name: Log In (rainy)	User Story #: 289
Description:	Verify that users cannot engage in Cross-Site Scripting using the log-in form.	

Pre-conditions:	1. User has successfully registered an account with GenomePro. 2. User has navigated to the GenomePro front page.
------------------------	--

Steps	Action	Value
1.	User enters a malicious username.	; DROP TABLE users;
2.	User enters their password.	p4ssword
3.	User clicks the "Login" button.	N/A
Result: ✓ Passed		

Post-conditions:	The User is presented with an error message indicating that his/her log-in credentials are invalid on a reloaded GenomePro front page.

Test case #: 7	Test case name: Submitting a Job (rainy)	User Story #: 291
Description:	Verify that users cannot enter malicious information into the additional arguments fields	

Pre-conditions:	1. User has logged into the system 2. User is on the Tools page
------------------------	--

Steps	Action	Value
1.	User selects the "Genome Probes" tool	Genome Probes
2.	User selects the a file for the tool	201602011319_fasta.txt
3.	User selects enters a malicious additional argument	; DROP TABLE users;
4.	User selects clicks the "Submit" button	N/A
Result: ✓ Passed		

Post-conditions:	The User is presented with a reloaded Tools page along with an error message indicating an invalid parameter for the tool selected.

Test case #: 8	Test case name: Submitting Multiple Jobs (sunny)	User Story #: 285
Description:	Verify that Users are truly able to submit multiple jobs	

Pre-conditions:	1. Users are logged into GenomePro 2. Users are on the Tools page
------------------------	--

Steps	Action	Value
1.	User selects two tools from the tool selection drop-down	(Repeated Seq., Unique Seq.)
2.	User selects two files from the file selection drop-down	(2016021.fafsa.txt, 2016022.fafsa.txt)
3.	User clicks the submit button	N/A
Result: ✓ Passed		

Post-conditions:	The User is presented with a reloaded Tools page along with a message indicating that the jobs were submitted for processing successfully.

Test case #: 9	Test case name: Submitting Multiple Jobs (rainy)	User Story #: 285
Description:	Verify that users are not, suddenly, able to submit jobs with erroneous text arguments	

Pre-conditions:	1. Users are logged into GenomePro 2. Users are on the Tools page
------------------------	--

Steps	Action	Value
1.	User selects two tools from the tool selection drop-down	(Repeated Seq., Unique Seq.)
2.	User selects two files from the file selection drop-down	(2016021.fafsa.txt, 2016022.fafsa.txt)
3.	User enters an erroneous argument	"erroneous_argument"
4.	User clicks the "Submit" button	N/A
Result: ✓ Passed		

Post-conditions:	The User is presented with a reloaded Tools page along with a message indicating that the argument entered is invalid

Test case #: 10	Test case name: Upload a File (sunny)	User Story #: 286
Description:	Verify that users are unable to able to upload files to GenomePro	

Pre-conditions:	1. User is logged into GenomePro 2. User is on the Tools page
------------------------	--

Steps	Action	Value
1.	User selects a file to upload using the built-in file browser	20160213.fafsa.txt
2.	User clicks the upload button	N/A
Result: ✓ Passed		

Post-conditions:	The User is presented with a reloaded Tools page displaying a message indicating that their file was uploaded to the system successfully

Test case #: 11	Test case name: Upload a File (rainy)	User Story #: 286
Description:	Verify that users are unable to upload corrupted or malicious files to the system	

Pre-conditions:	1. User is logged into GenomePro 2. User is on the Tools page
------------------------	--

Steps	Action	Value
1.	User selects a corrupted file to upload using the file browser	corrupted_file.png
2.	User clicks the upload button	N/A
Result: ✓ Passed		

Post-conditions:	The User is presented with a reloaded Tools page displaying a message indicating that their file was unable to be uploaded to the system and asking the user to please try again with a different file

Test case #: 12	Test case name: Login-Sunny	User Story #: 287
Description:	Verify login feature entering correct user name and password to show successful login add event to the table	

Pre-conditions:	User is on home page
------------------------	----------------------

Steps	Action	Value
1	Type user correct user name	Rarci003
2	Type correct password	123456
3	Click login button	ClickAndWait
4	Verify Text	You have successfully logged in!
5	Verify record on usage table	15, timestamp
Result: ✓Passed		

Post-conditions:	User is logged in and the usage table has a new record with the user id and the timestamp of the login event.
-------------------------	---



Test case #: 13	Test case name: Login-Rainy	User Story #: 233
Description:	Verify login feature entering incorrect user name and/or password to show unsuccessful login	

Pre-conditions:	User is on home page
------------------------	----------------------

Steps	Action	Value
1	Type user incorrect user name	rarci005
2	Type correct password	123456
3	Click login button	ClickAndWait
4	Verify Text	The credentials given are incorrect. Please try again.
Result: ✓Passed		

Post-conditions:	User is redirected to Home page, no records add to the usage table.

Test case #: 14	Test case name: Create new reply	User Story #: 294
Description:	Verify new reply post	

Pre-conditions:	User is on home page and logged
------------------------	---------------------------------

Steps	Action	Value
1	Select a topic	
2	Type reply to post	Thios is a test reply
3	Click submit button	
4	Verify reply	
Result: ✓Passed		

Post-conditions:	Home page shows topic with the reply

Test case #: 15	Test case name: Create new topic	User Story #: 294
Description:	Verify create new topic feature	

Pre-conditions:	User is on home page and logged in GenomePro
-----------------	--

Steps	Action	Value
1	Click Create Topic	
2	Type topic title	This is a test
3	Type topic body	Hello, this is a test
4	Click Submit	
5	Verify new topic on home page	
Result: ✓Passed		

Post-conditions:	Home page shows new topic

Test case #: 1i	Test case name: Submitting a Job (integration)	User Story #: 282
Description:	Verify that submitting a job interacts correctly with the other subsystems	

Pre-conditions:	1. User is logged into the system 2. User is on the Tools page
------------------------	---

Steps	Action	Value
1.	User sets up a job for submission	(Unique Seq., 20160211.fasta.txt.sorted)
2.	User clicks the "Submit" button	N/A
3.	User verifies their job submission in the Job History table	N/A
4.	User clicks on the "Profile" link	N/A

Result: ✓ Passed

Post-conditions:	User is confronted with the correct and unaltered Profile page, having previously verified that the job was successfully added to the job history table on the tools page.

Test case #: 2i	Test case name: Submitting Multiple Jobs (integration)	User Story #: 282
Description:	Verify that submitting a job interacts correctly with the other subsystems	

Pre-conditions:	1. User is logged into the system 2. User is on the Tools page
------------------------	---

Steps	Action	Value
1.	User sets up a single job for submission	(Unique Seq., 20160211.fasta.txt.sorted)
2.	User clicks the "Submit" button	N/A
3.	User verifies their job submission in the Job History table	N/A
4.	User clicks on the "Profile" link	N/A

Result: ✓ Passed

Post-conditions:	User is confronted with the correct and unaltered Profile page, having previously verified that the job was successfully added to the job history table on the tools page.

Test case #: 3i	Test case name: Uploading a File (integration)	User Story #: 286
Description:	Verify that the functionality for uploading a file does not hinder the other subsystems of GenomePro	

Pre-conditions:	1. User is logged into the system 2. User is on the Tools page
------------------------	---

Steps	Action	Value
1.	User selects a file for uploading	20160214.fafsa.txt
2.	User clicks the "Submit" button	N/A
3.	User verifies their file was successfully uploaded	N/A
4.	User clicks on the "Profile" link	N/A
Result: ✓ Passed		

Post-conditions:	User is confronted with the correct and unaltered Profile page, having previously verified that the file was successfully added to GenomePro

Test case #: 4i	Test case name: Update Profile (integration)	User Story #: 288
Description:	Verify that the functionality for updating a User's profile does not negatively impact the rest of the subsystems	

Pre-conditions:	1. User is logged into GenomePro 2. User is on their Profile page
------------------------	--

Steps	Action	Value
1.	User enters a new email address into the Profile Update form	samuraifamploo@gmail.com
2.	User enters their password (twice) into the Profile Update form	л4ссц0Рд
3.	User clicks the Update button	N/A
4.	User clicks on the " Tools " link	N/A
Result: ✓ Passed		

Post-conditions:	The User is confronted with the correct and unaltered Tools page, having previously verified that their profile was successfully updated

Test case #: 5i	Test case name: Log In (integration)	User Story #: 289
Description:	Verify that the functionality for logging in does not interfere with any other subsystem	

Pre-conditions:	1. User has a registered account on GenomePro 2. User is on the home page for the GenomePro website
------------------------	--

Steps	Action	Value
1.	User enters his username into the appropriate field	dgonz293
2.	User enters his password into the appropriate field	л4ссщ0Рд
3.	User clicks the "Log In" button	N/A
4.	User clicks on the " Tools " link	N/A
Result: ✓ Passed		

Post-conditions:	The User is confronted with the correct and unaltered Tools page

Test case #: 6i	Test case name: Submit Multiple Jobs (integration)	User Story #: 291
Description:	Verify that the new job submission algorithm does not impact the functionality of GenomePro's other subsystems	

Pre-conditions:	1. User is logged into the system 2. User is on the Tools page
------------------------	---

Steps	Action	Value
1.	User sets up a job for submission	(Unique Sequences, 201614.fafsa.txt)
2.	User clicks the "Submit" button	N/A
3.	User clicks the "Profile" link	N/A
4.	User verifies their job submission in their Job History table	N/A
Result: ✓ Passed		

Post-conditions:	The User is confronted with the correct Profile page, along with a new entry in their Job History table detailing the job that was just submitted for processing

GLOSSARY

CLASS DIAGRAM	Describes the system in terms of <i>classes</i> , <i>attributes</i> , <i>operations</i> , and their relationships.
CRON	Linux Utility that schedules a command or script on a server to run automatically on a specified time and data.
DARK MATTER	Genetic material that are unclassified or not well understood.
FASTA	File type extension given to genomic data.
FASTQ	File type extension given to genomic data.
GENOME SIGNATURES	A tool for finding differences amongst sets of genomes
JOB	The process of submitting files as input for tools with the expectation of getting specific result files.
MAP	An ordered list of repeated sub-sequences extracted from a genome file.
MODEL	Abstraction of reality.
MEDIA QUERY	CSS module allowing content rendering to adapt to conditions such as screen resolution (e.g. smartphone screen vs. computer screen).
SEQUENCER	A Professional in the biomedical research field that closely works with genome sequences.
SEQUENCE DIAGRAM	A sequence diagram is an interaction diagram that emphasizes the time ordering of events and messages.
SERVER	A server is a system containing both hardware and software that can provide services to the client machines, such as file access, printing, archiving, or database access.
PROBE	A sub-sequence extracted from a genome sequence.
USER	External entity that interact with the system.
USER INTERFACE	The way through which a user interacts with the computer system.
VALIDATOR	Term used to define a program that validates the integrity of a genome file and displays its data type, e.g. DNA, RNA or Protein.

APPENDIX

Appendix A - UML Diagrams

Static UML Diagrams

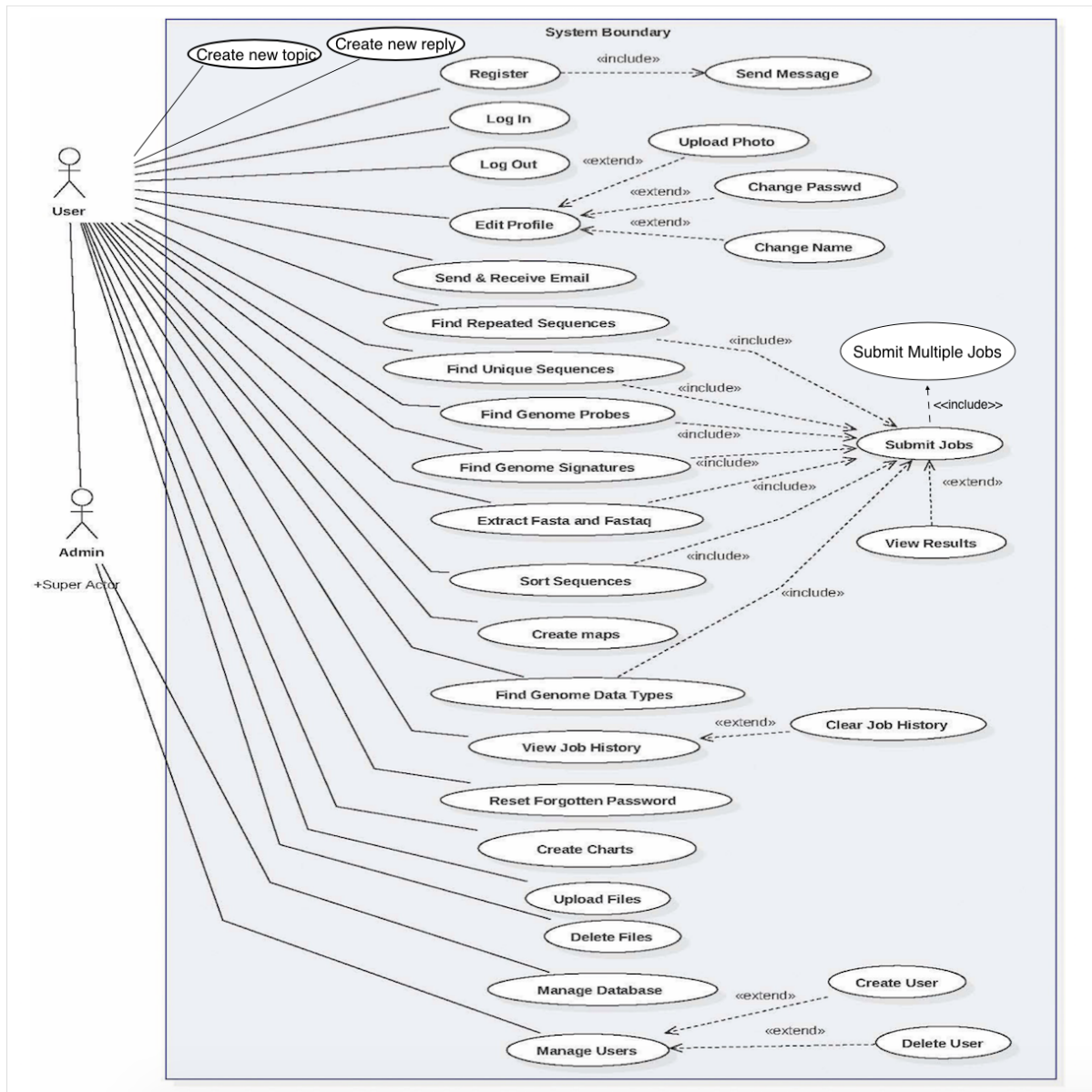


Figure 3 - Use Cases Diagram

Database Modeling Diagram

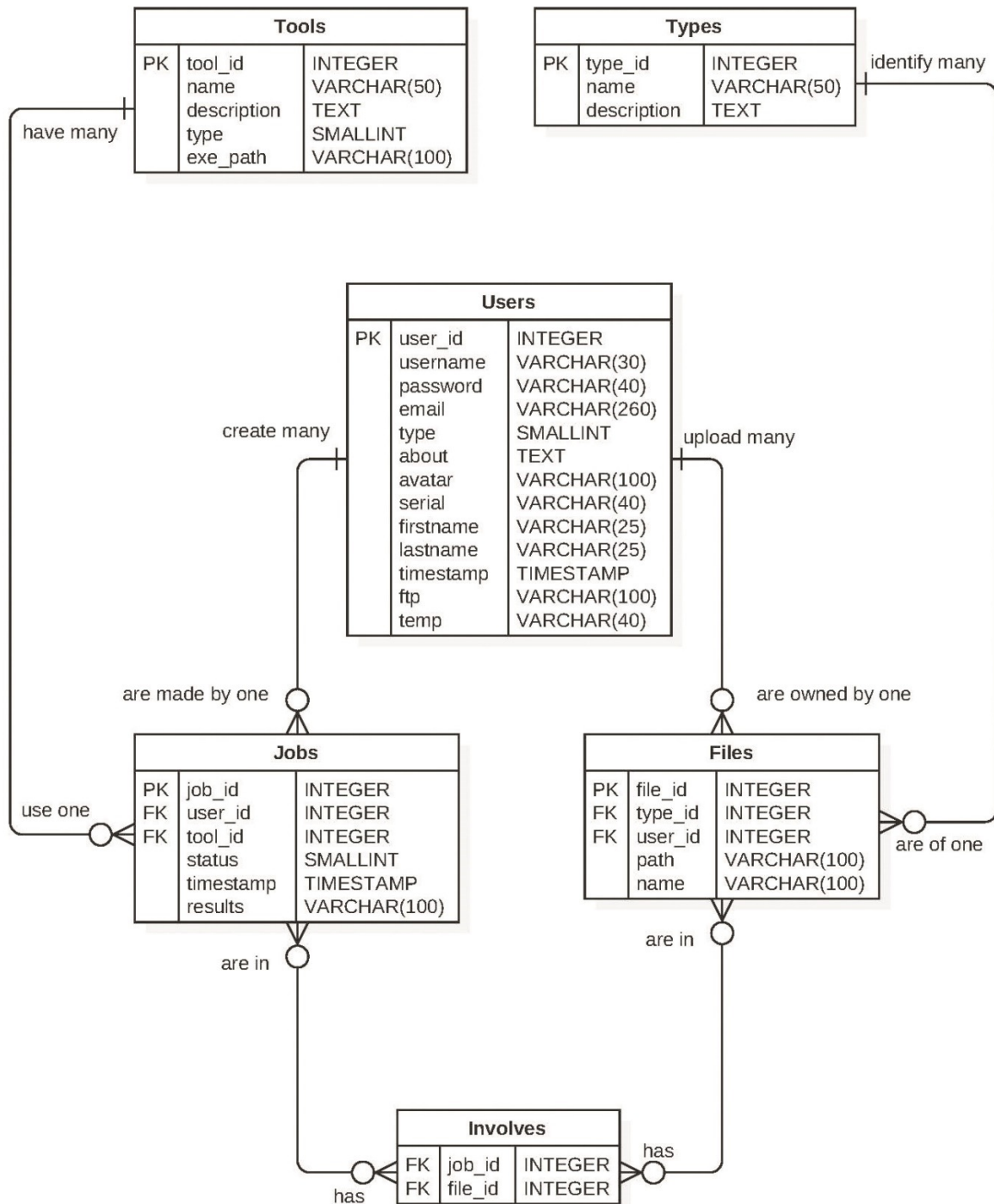


Figure 4 - Database Modeling Diagram

Class Diagram

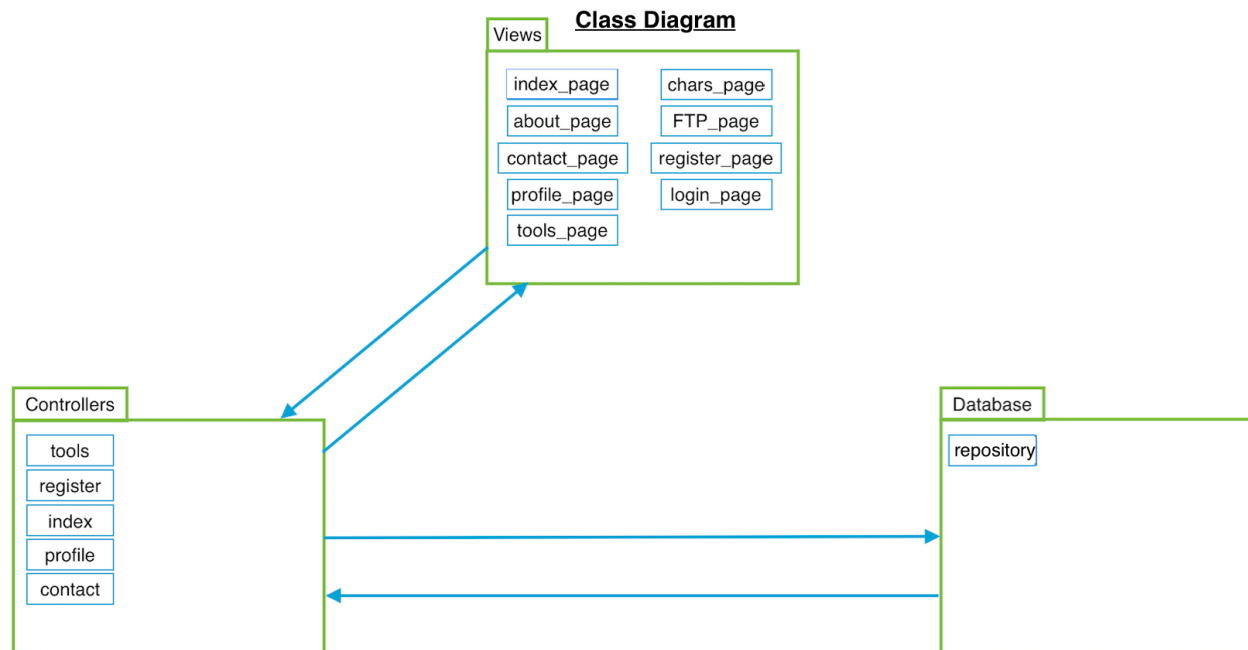


Figure 5 - Class Diagram

Model-Controller Diagram 1

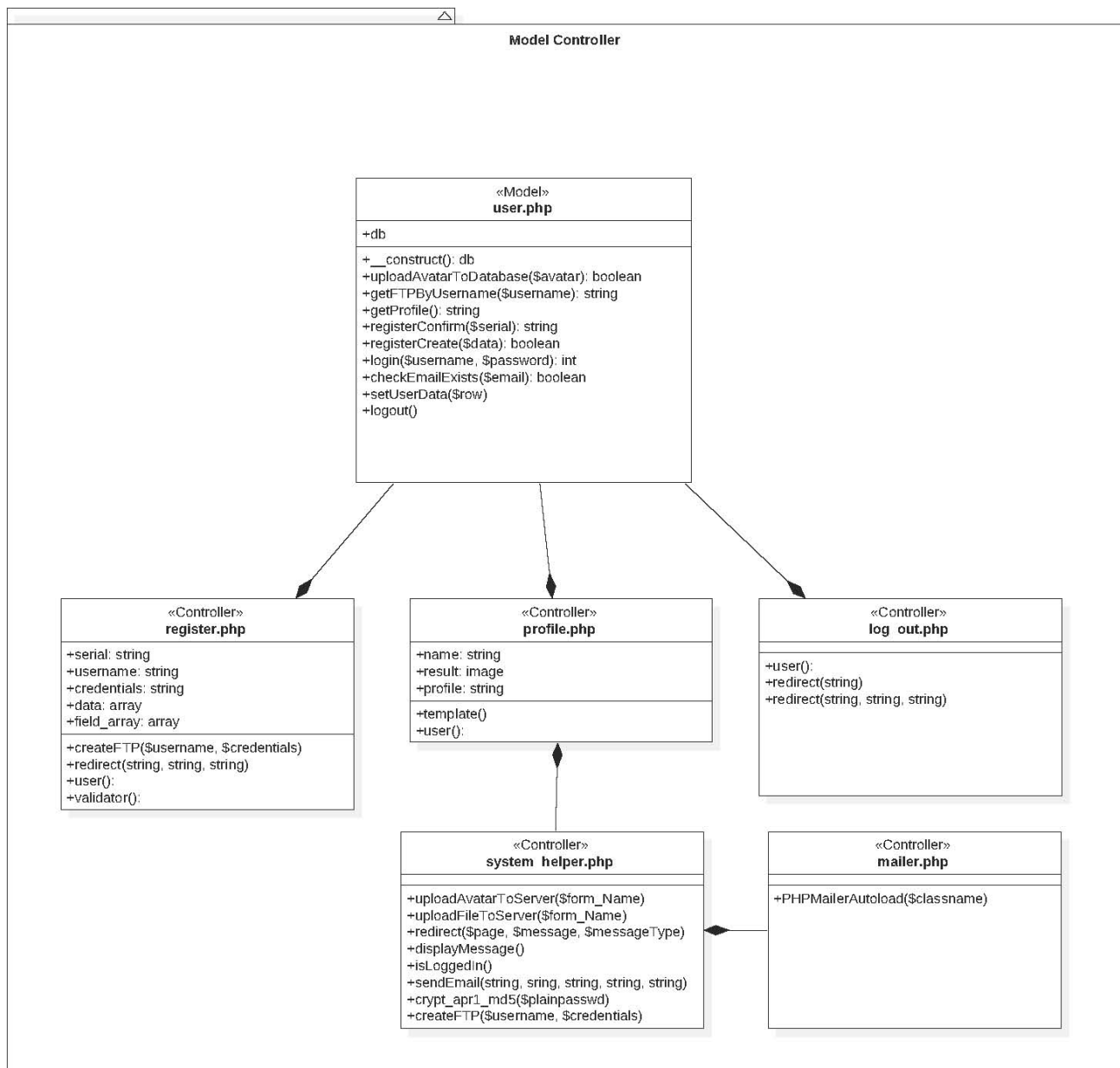


Figure 6 – Model-Controller Diagram 1

Model-Controller Diagram 2

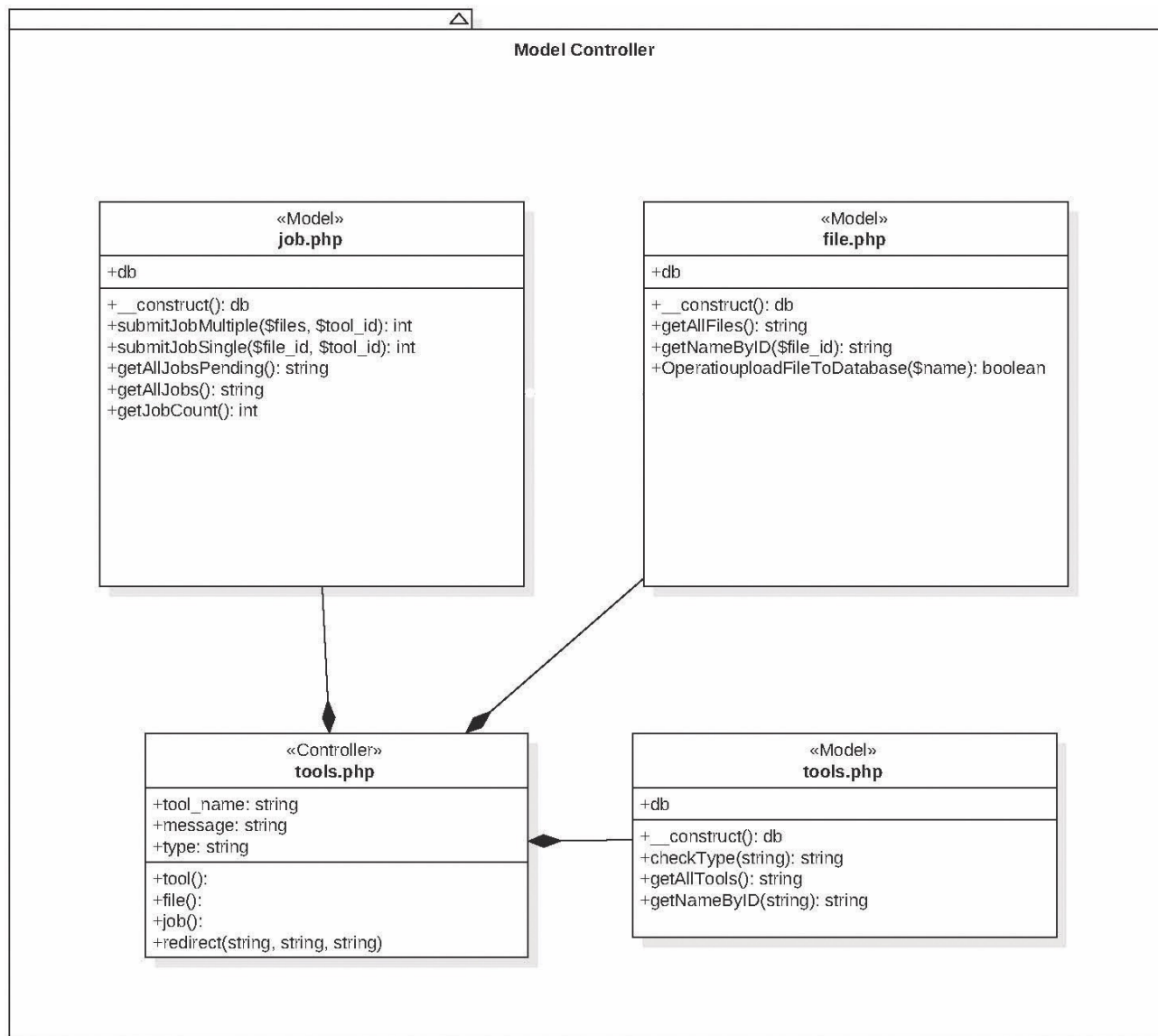


Figure 7 – Model-Controller Diagram

Model-Controller Diagram 3

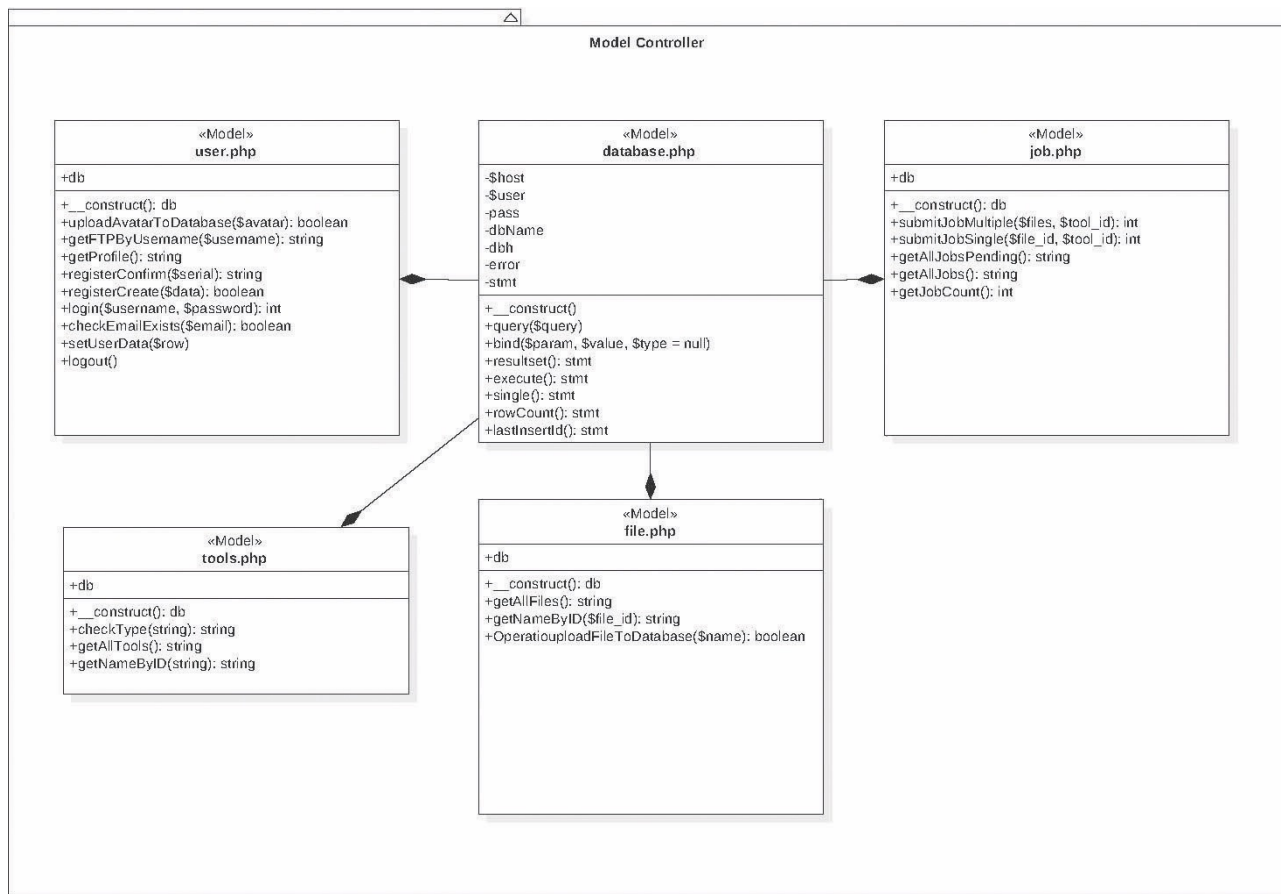


Figure 8 – Model-Controller Diagram 3

View-Controller Diagram 1

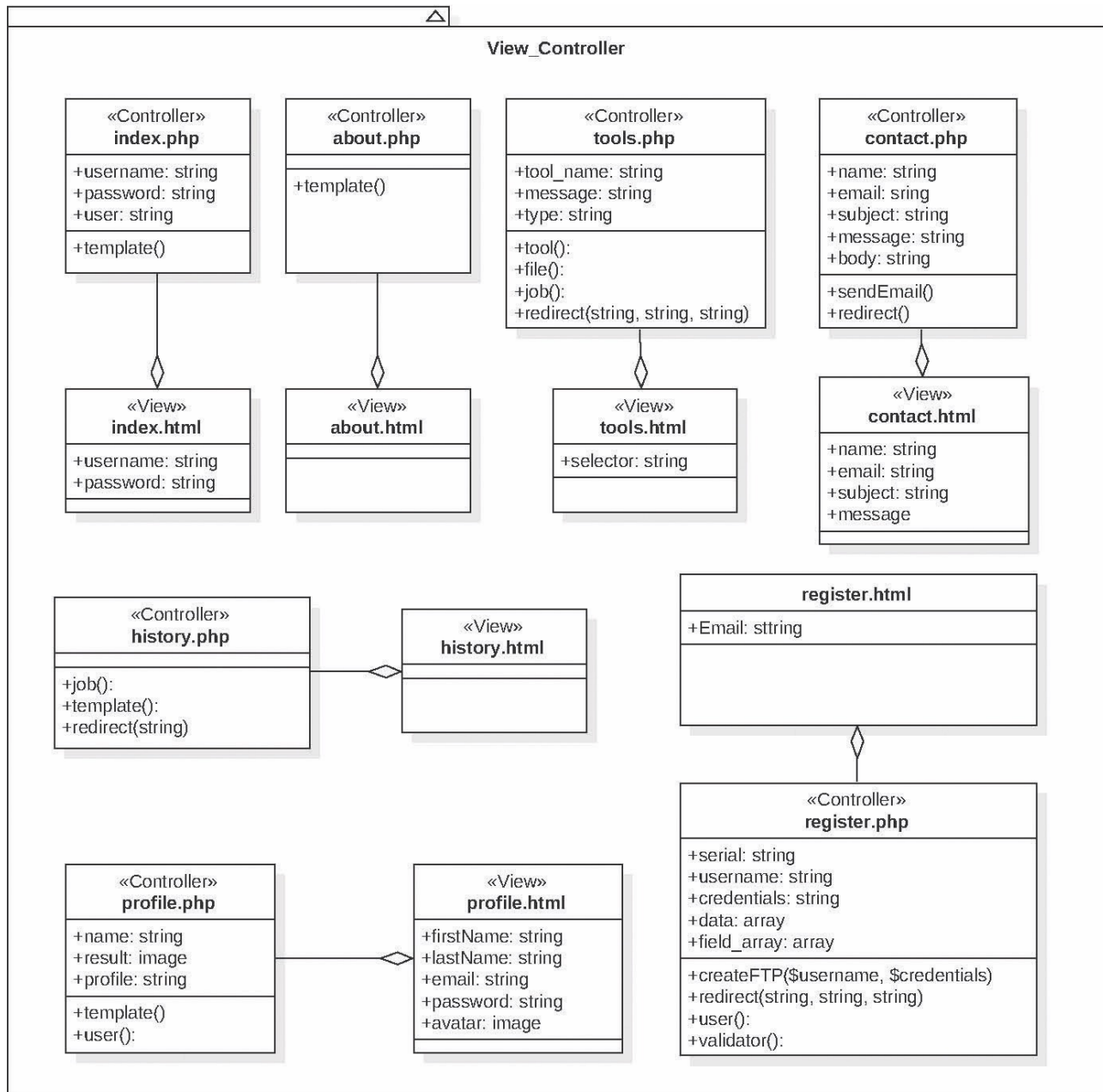


Figure 9 – View-Controller Diagram 1

View-Controller Diagram 2

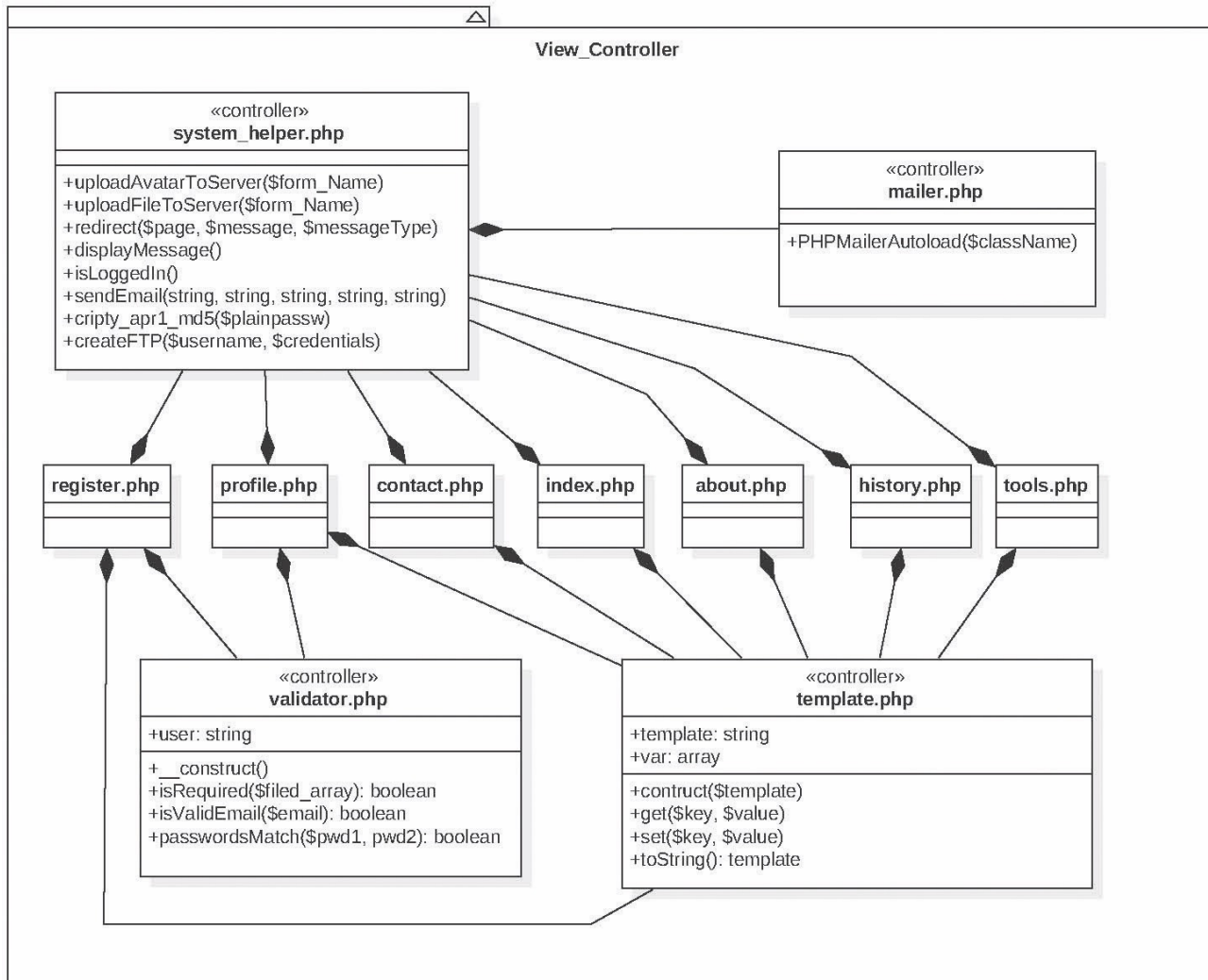


Figure 10 – View-Controller Diagram 2

Dynamic UML (Sequence) Diagrams

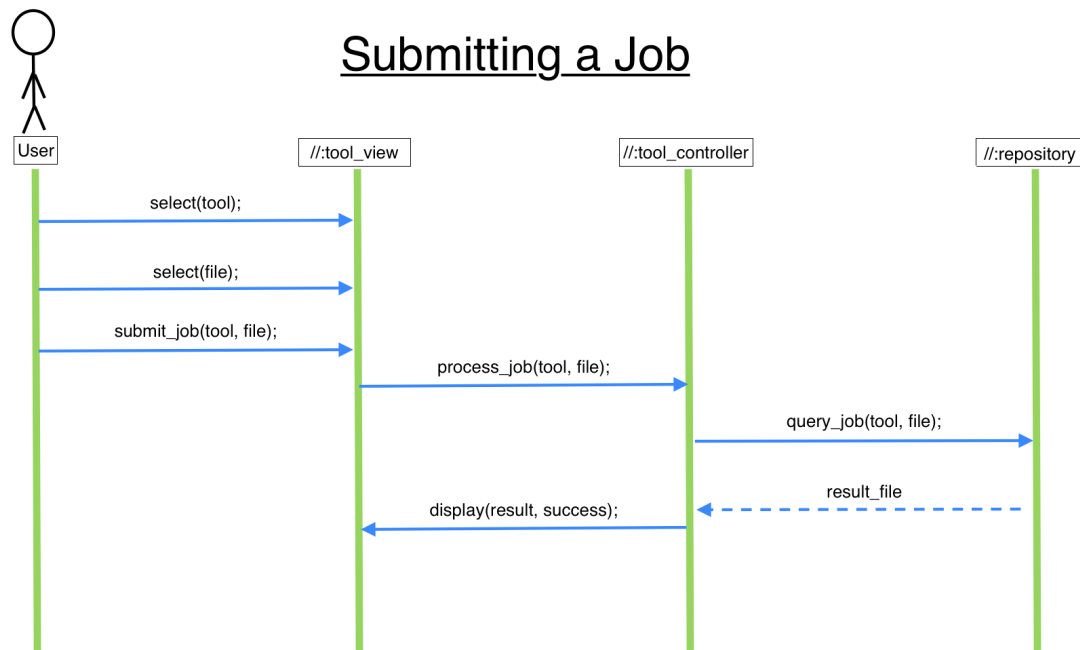


Figure 11 – Use Case #1 Sequence Diagram

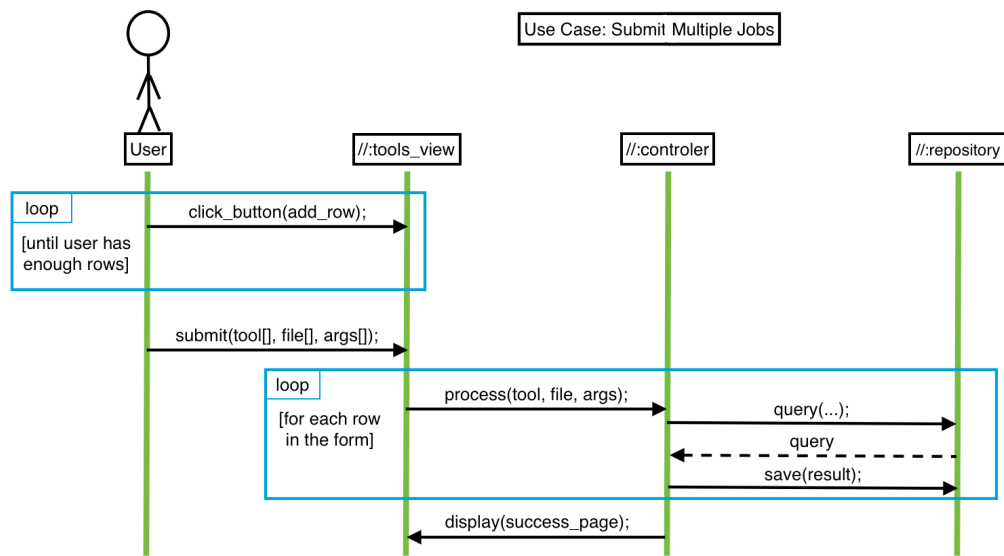


Figure 12 – Use Case #2 Sequence Diagram

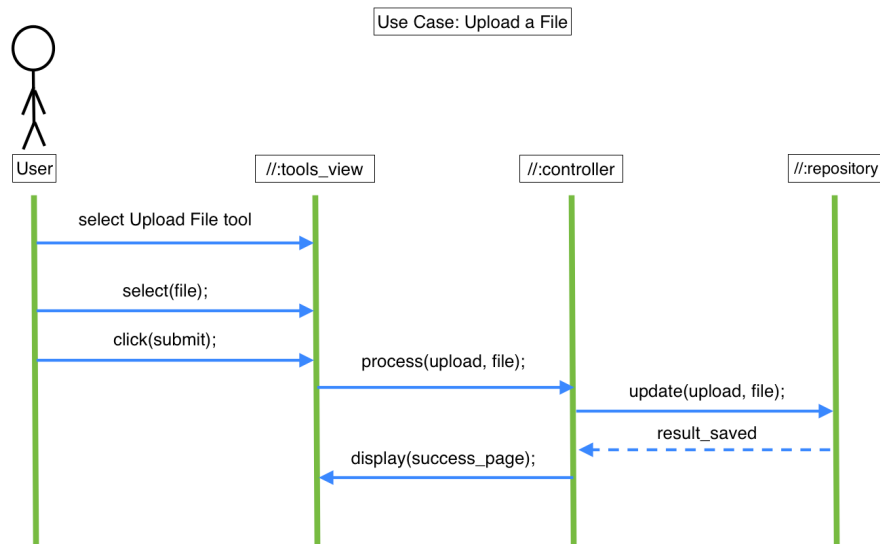


Figure 13 – Use Case #3 Sequence Diagram

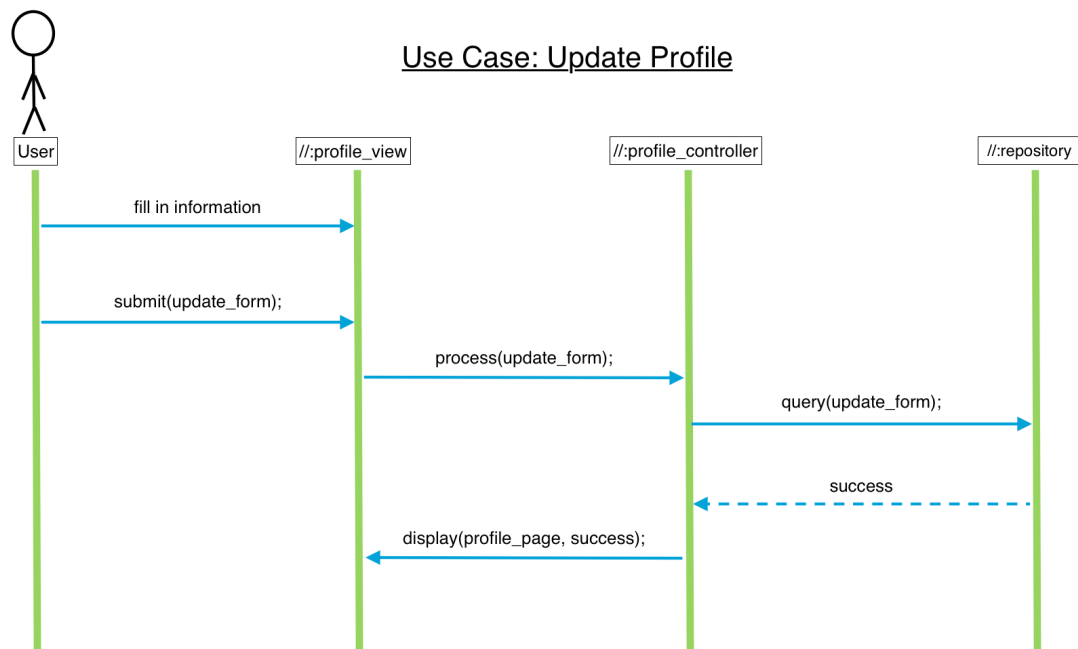


Figure 14 – Use Case #4 Sequence Diagram

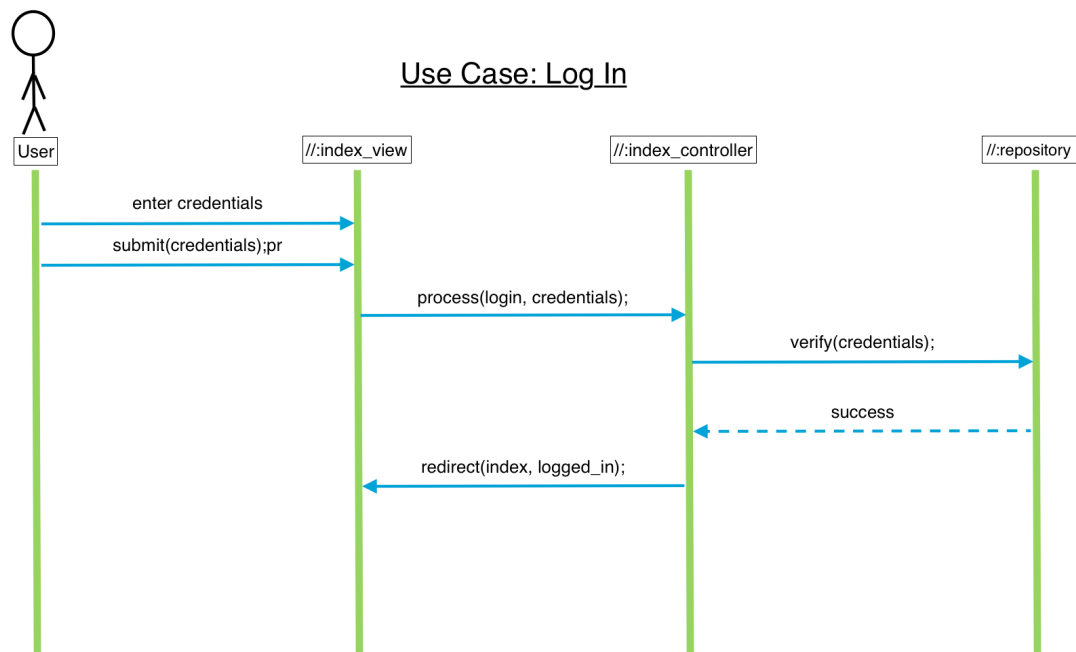


Figure 15 – Use Case #5 Sequence Diagram

Reply to a Topic

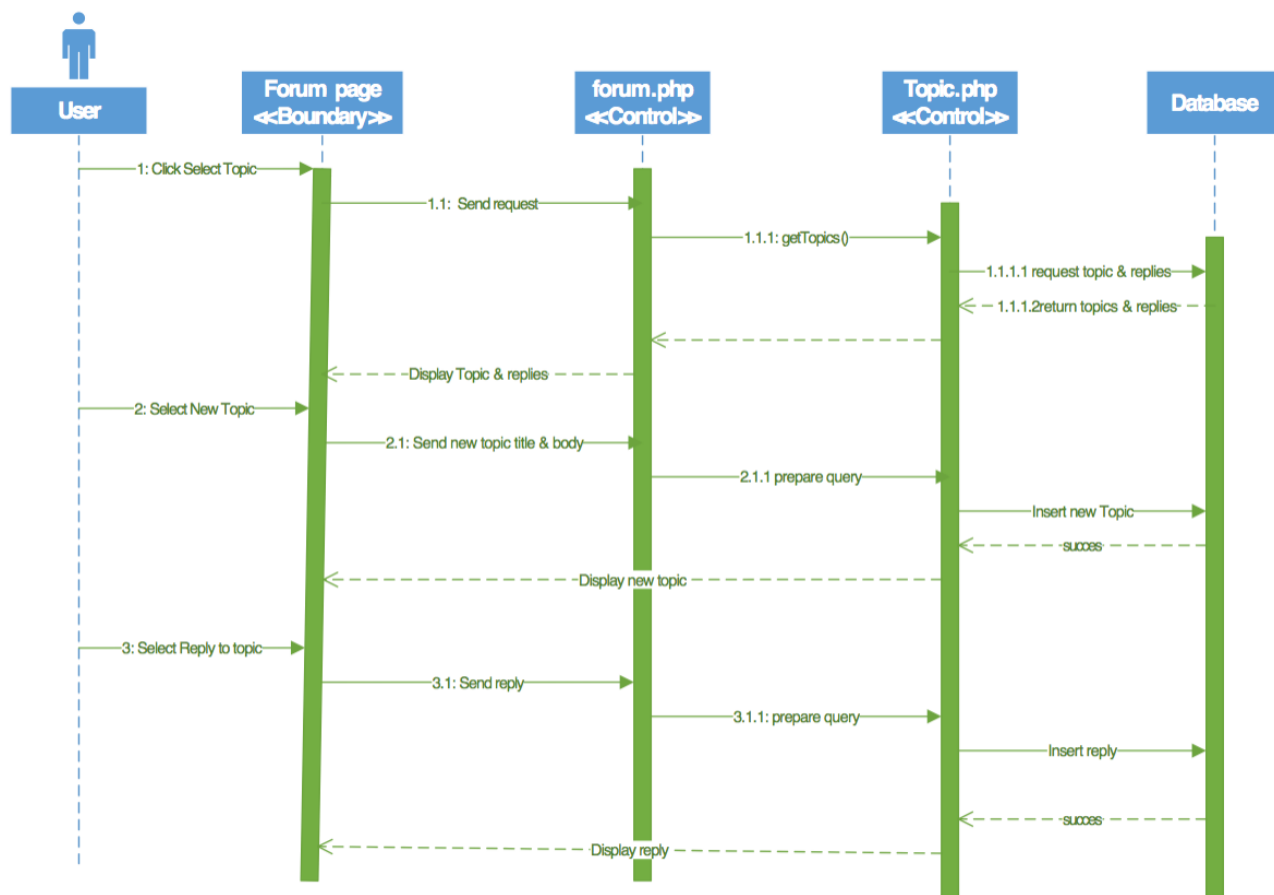


Figure 16 – Use Case #6 Sequence Diagram

sd Usage report page

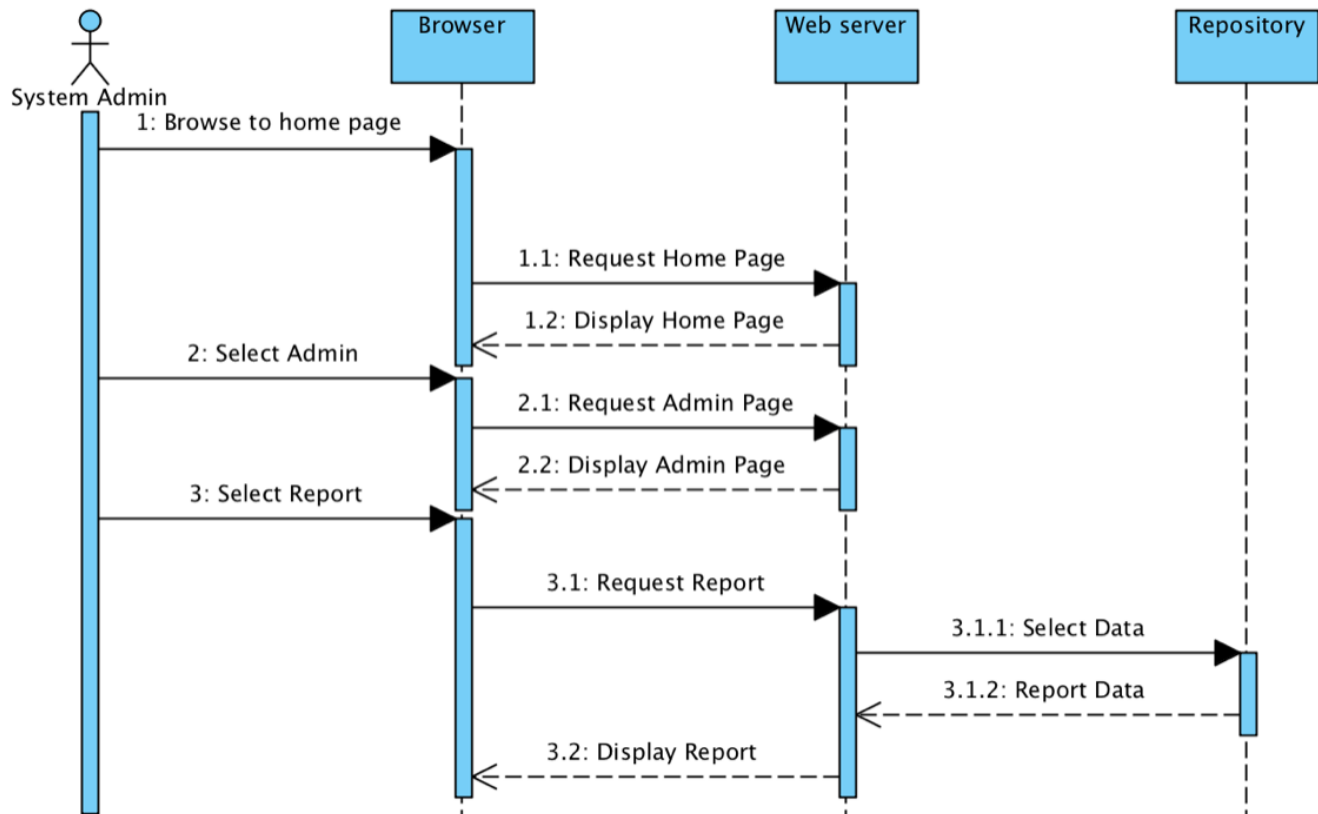


Figure 17 – Use Case #7 Sequence Diagram

Appendix B - User Interface Design

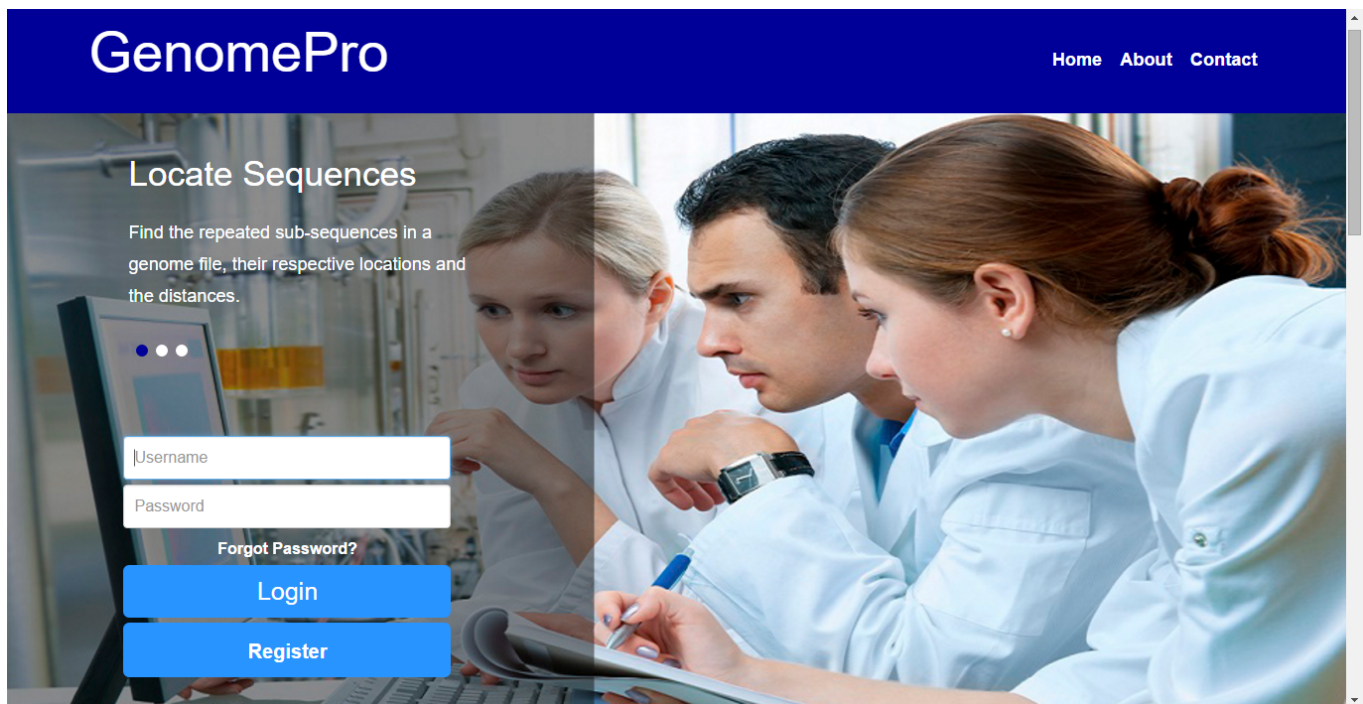


Figure 18 - Home Page Login Area

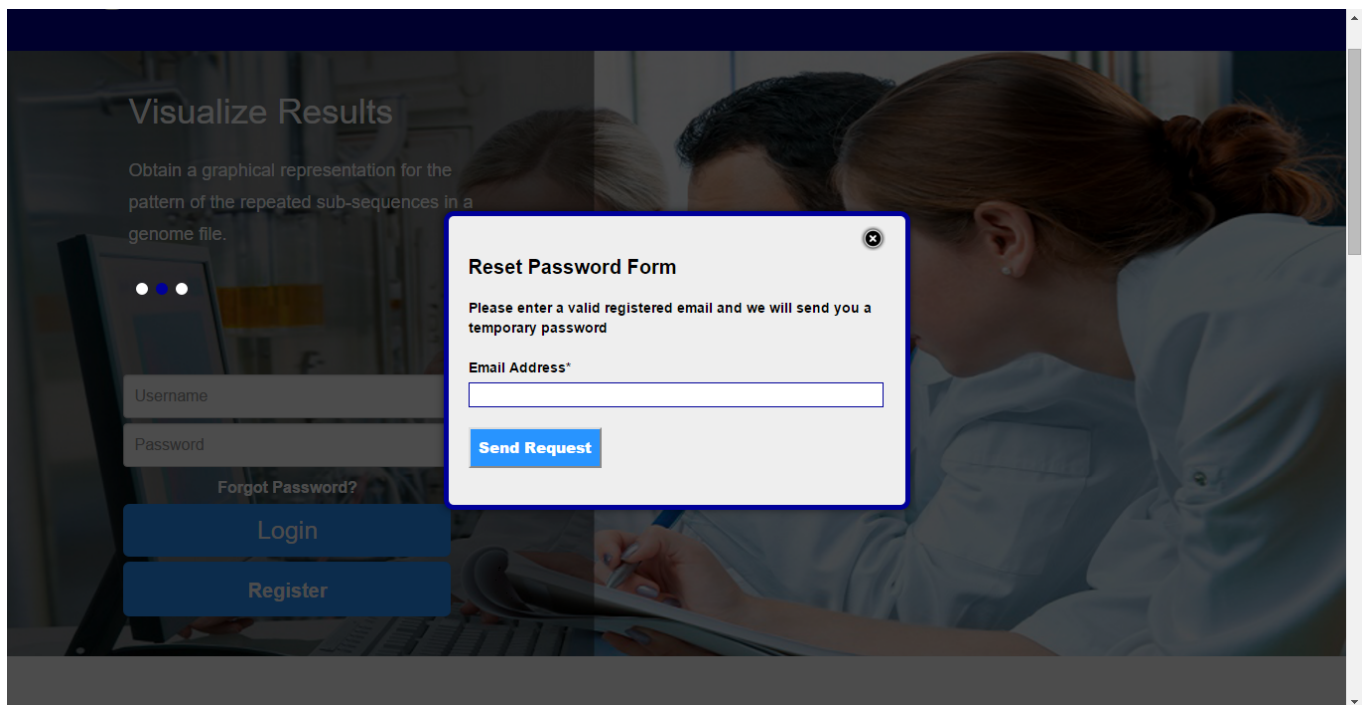
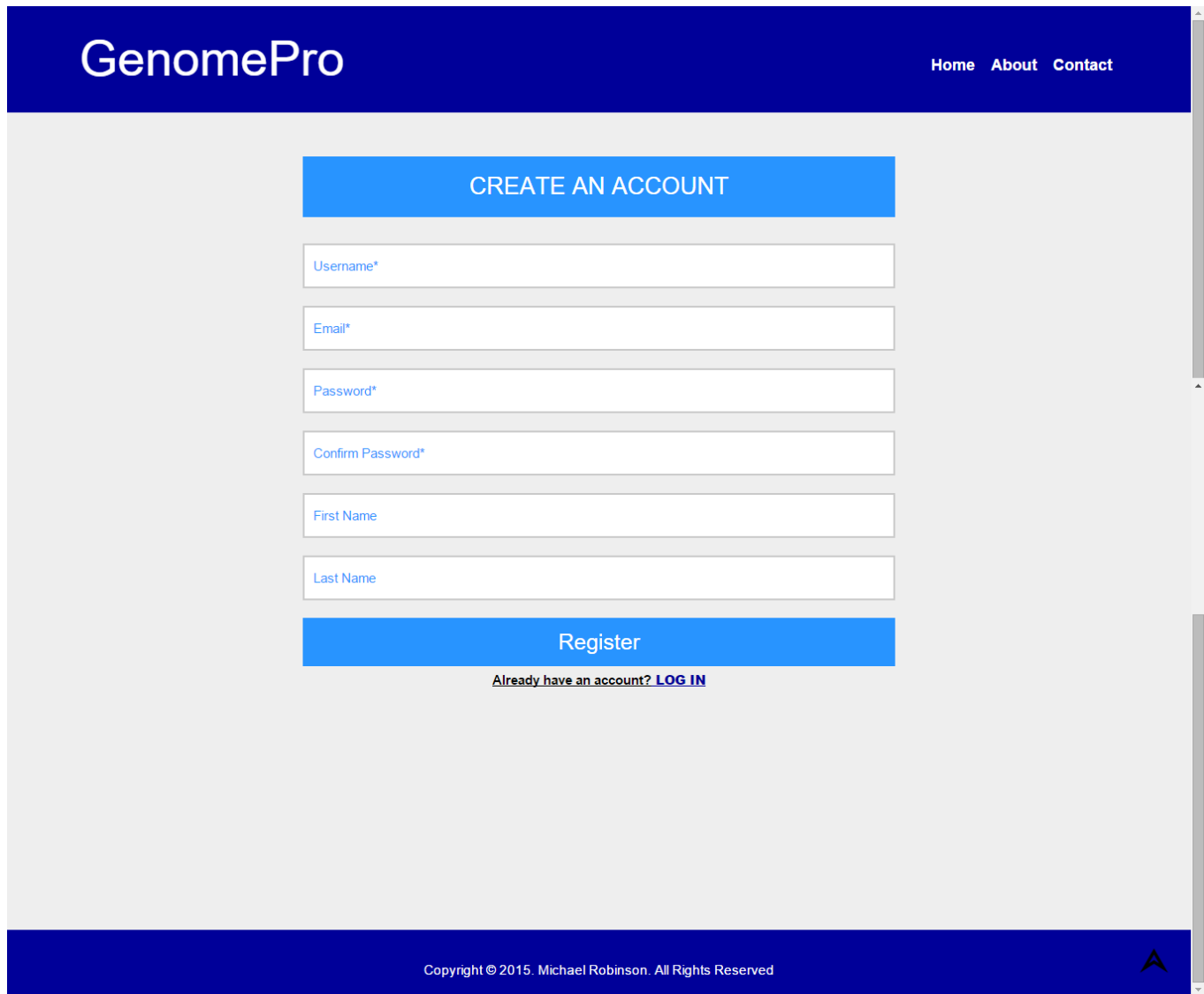


Figure 19 - Home Page - Forget Password Feature



The screenshot shows the 'Register' page of the GenomePro application. The page has a dark blue header with the 'GenomePro' logo on the left and navigation links 'Home', 'About', and 'Contact' on the right. The main content area is light gray and contains a registration form. At the top of the form is a blue button labeled 'CREATE AN ACCOUNT'. Below this are six input fields: 'Username*', 'Email*', 'Password*', 'Confirm Password*', 'First Name', and 'Last Name'. The asterisk indicates required fields. At the bottom of the form is a blue button labeled 'Register'. Below the 'Register' button is a link that says 'Already have an account? [LOG IN](#)'. The footer of the page is dark blue and contains the copyright text 'Copyright © 2015. Michael Robinson. All Rights Reserved' on the left and a small logo on the right.

GenomePro

Home About Contact

CREATE AN ACCOUNT

Username*

Email*

Password*

Confirm Password*

First Name

Last Name

Register

Already have an account? [LOG IN](#)

Copyright © 2015. Michael Robinson. All Rights Reserved

Figure 20 - Register Page

GenomePro

Home About Contact Profile Tools Charts FTP Logout

Logged in as: mmesq001

GenomePro at a Glance



Finding Signatures

Creation of genome finger prints/signatures that are predictive and prognostic for specific biologic events or treatment outcomes.



Comparison

We offer our users the power to efficiently process genomic data files and to create results containing accurate information about the differences, similarities, and the analytics between genome files



Preventive Health Care Measures

Identification of sources and types of bacteria or viruses that spread in hospital or community settings.



Cancer Risk Prediction And Assessment

Confirmation of the presence of either new primary lesions or metastatic disease in all solid tumors, aiding in staging of cancer and treatments strategies.



Tracking Mutations

Tracking of HIV/TB mutational changes in patients, offering the possibility for new treatment paradigms.



New Diseases Cure And Prevention

Creating genetic finger prints libraries of known bacteria, viruses, and tumors, which are currently cured with known treatments or medications, will allow us to search for those finger prints in other genomes and apply the known cures to new diseases. These finger prints can also be useful to identify the source of drug resistance.

Our Audience



What Professionals Say About GenomePro



Michael Robinson

The ability to find similarities in multiple genomes is a tremendous tool offered by GenomePro. Not only you are able to visualize results with graphical comparisson but you can also find out the presence of unique and/or repeated sub-sequences in your files to perform more efficient and robust analysis.

Copyright © 2015. Michael Robinson. All Rights Reserved



Figure 21 - Home Page for Logged In Users

May 5th, 2016

Page 58 of 77

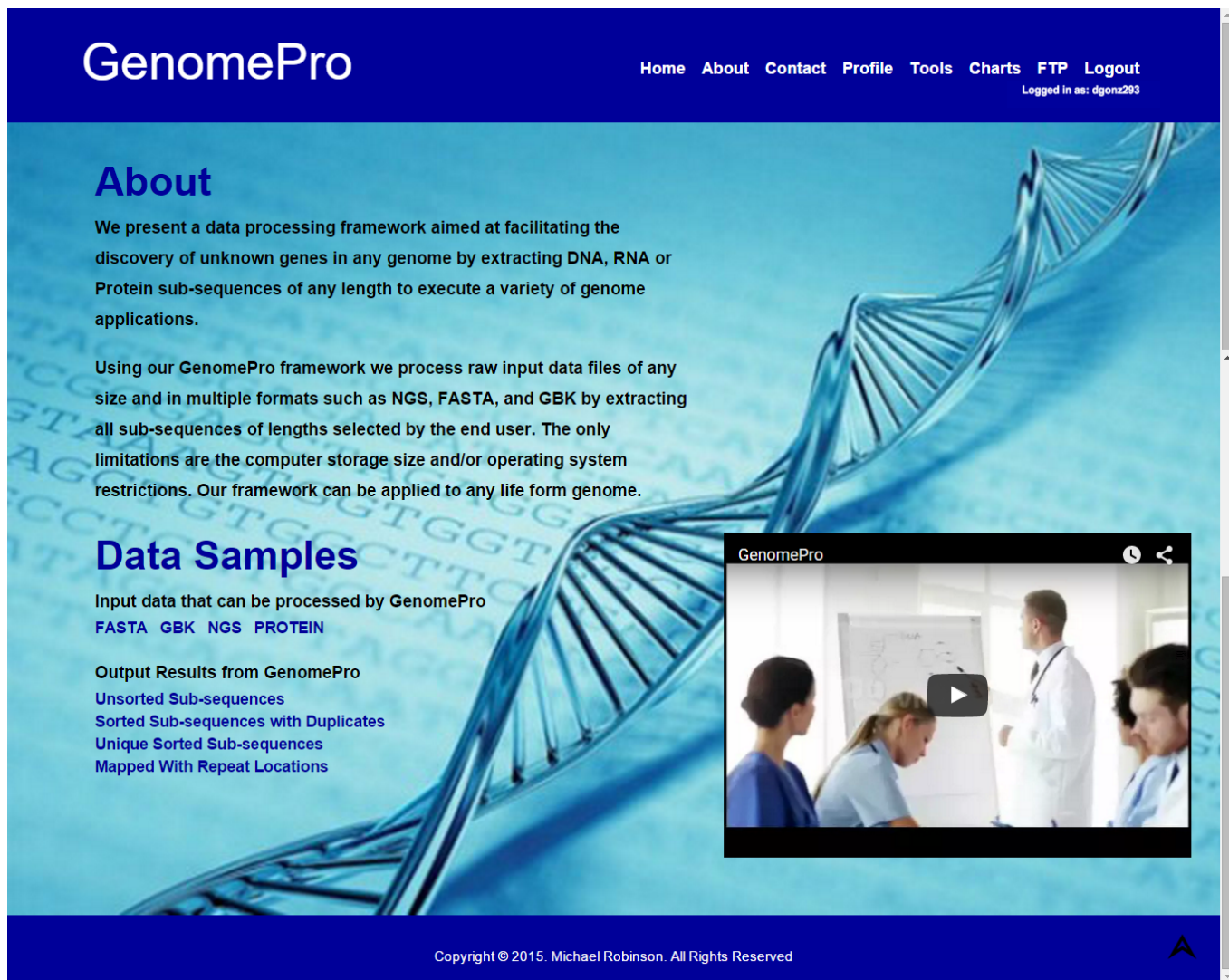


Figure 22 - About Page

GenomePro

[Home](#) [About](#) [Contact](#) [Profile](#) [Tools](#) [Charts](#) [FTP](#) [Logout](#)

Logged in as: dgonz293

Contact Us

Contact Form

NAME *

EMAIL ADDRESS *

SUBJECT *

MESSAGE *

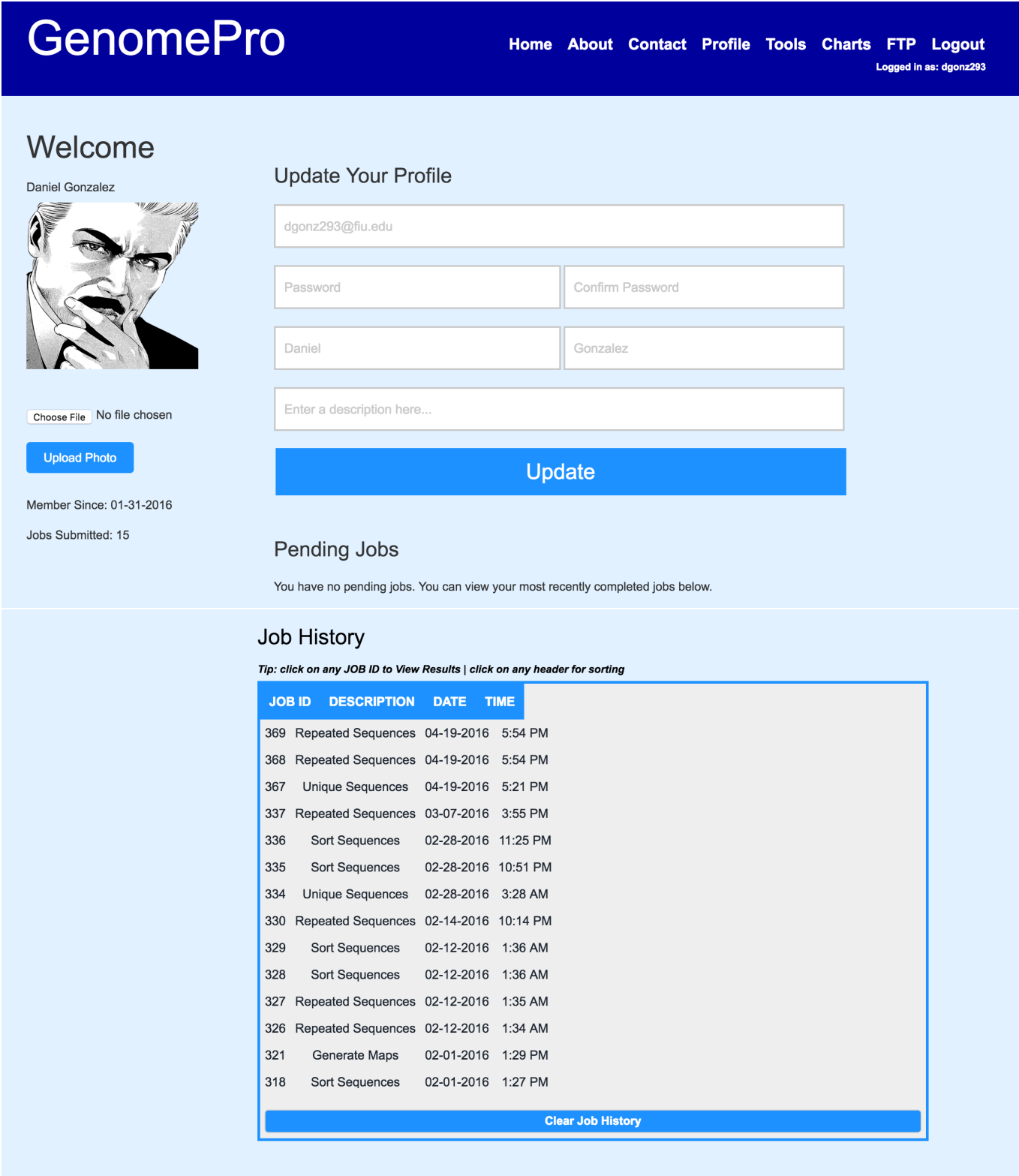
Send

Address

Florida International University,
Modesto Campus - Miami
USA
Phone:305.999.9999
Fax: 305.888.8888

Copyright © 2015. Michael Robinson. All Rights Reserved

Figure 23 - Contact Page



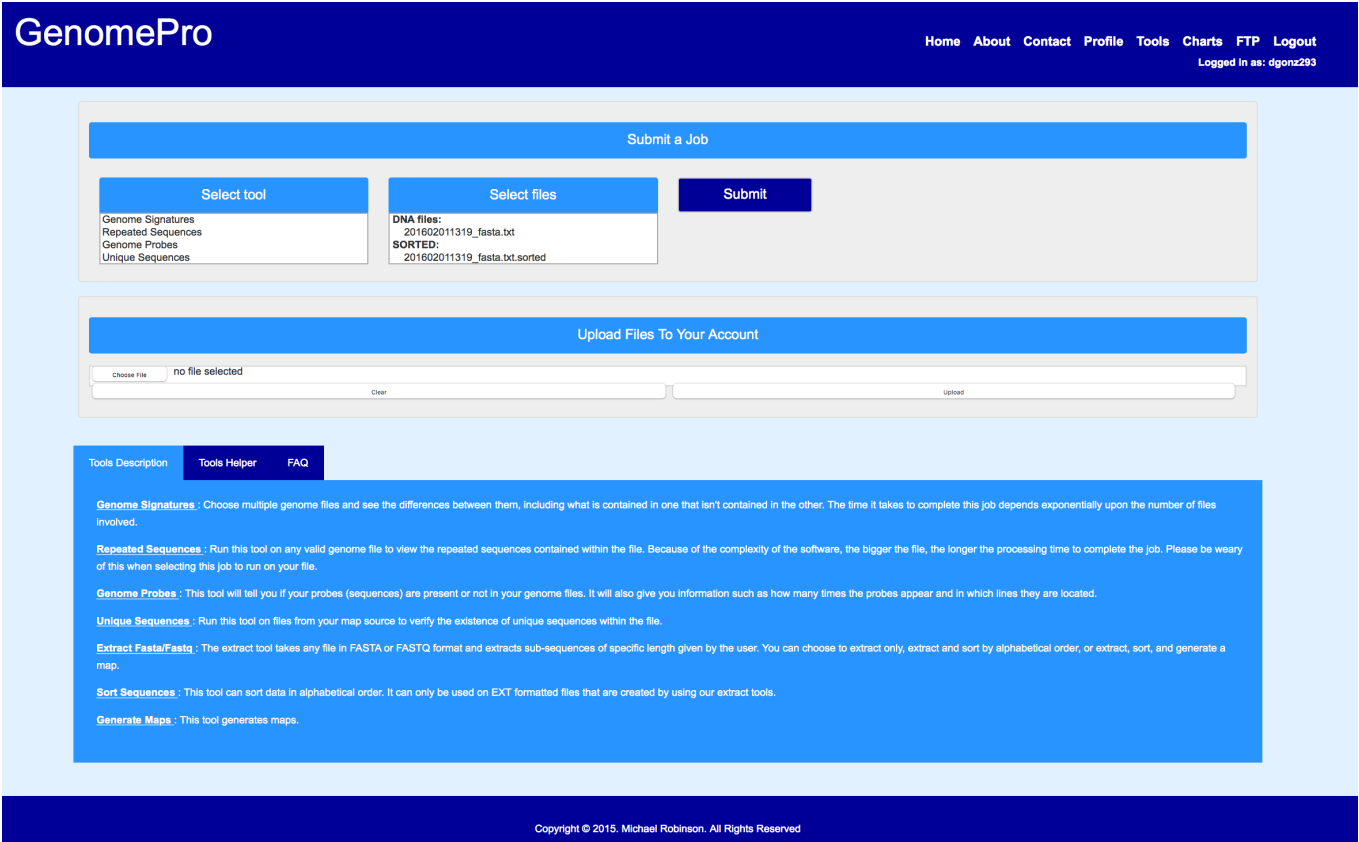


Figure 25 - Tools Page for logged in users

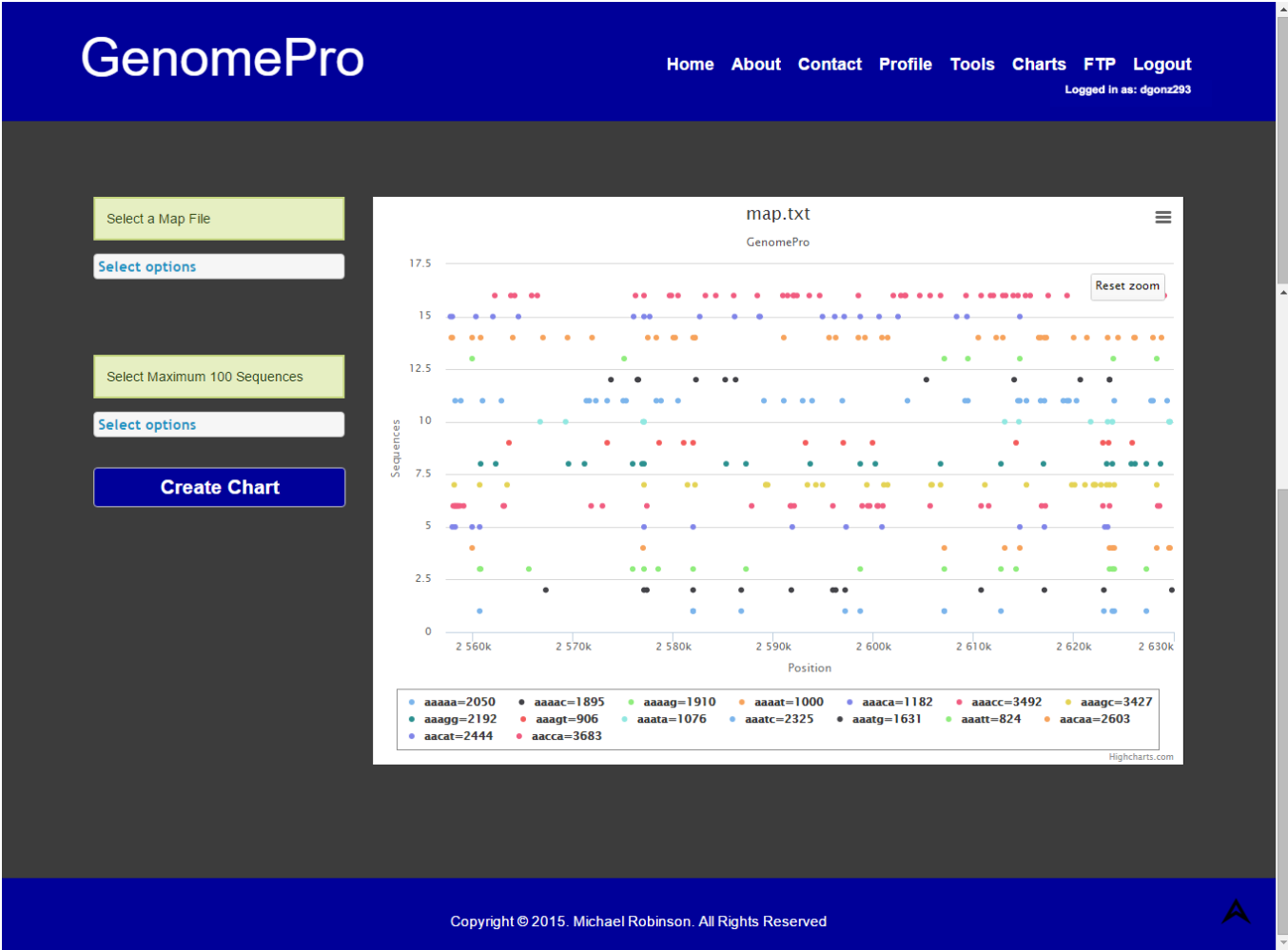


Figure 26 - Charts Page for logged in users

Job History

Tip: click on any JOB ID to View Results | click on any header for sorting

JOB ID	DESCRIPTION	DATE	TIME
369	Repeated Sequences	04-19-2016	5:54 PM
368	Repeated Sequences	04-19-2016	5:54 PM
367	Unique Sequences	04-19-2016	5:21 PM
337	Repeated Sequences	03-07-2016	3:55 PM
336	Sort Sequences	02-28-2016	11:25 PM
335	Sort Sequences	02-28-2016	10:51 PM
334	Unique Sequences	02-28-2016	3:28 AM
330	Repeated Sequences	02-14-2016	10:14 PM
329	Sort Sequences	02-12-2016	1:36 AM
328	Sort Sequences	02-12-2016	1:36 AM
327	Repeated Sequences	02-12-2016	1:35 AM
326	Repeated Sequences	02-12-2016	1:34 AM
321	Generate Maps	02-01-2016	1:29 PM
318	Sort Sequences	02-01-2016	1:27 PM

Clear Job History

Figure 27 - FTP Page for logged in Users

Appendix C - Sprint Planning Reports

Sprint 1 Planning Report

Date: January 29th, 2016

Attendees: Daniel Gonzalez, Roberto Arciniegas, Guido Ruiz, Professor Robinson

Discussed Topics: After discussion, the velocity of the team were estimated to be 40 story points. The product owner chose the following user stories to be done during the next sprint. They are ordered based on their priority.

- User Story #280: Prerequisites/Acquaintance with Pre-existing Software Solution
- User Story #281: Prerequisites/Acquaintance with existing software

The team members indicated their willingness to work on the following user stories.

- Daniel Gonzalez: User Story #280
- Roberto Arciniegas: User Story #281

Sprint 2 Planning Report

Date: February 12th, 2016

Attendees: Daniel Gonzalez, Roberto Arciniegas, Guido Ruiz, Professor Robinson

Discussed Topics: After discussion, the velocity of the team were estimated to be 40 story points. The product owner chose the following user stories to be done during the next sprint. They are ordered based on their priority.

- User Story #282: Redesign Tools Page (front end)
- User Story #283: FTP file manager plug in

The team members indicated their willingness to work on the following user stories.

- Daniel Gonzalez: User Story #282
- Roberto Arciniegas: User Story #283

Sprint 3 Planning Report

Date: February 26th, 2016

Attendees: Daniel Gonzalez, Roberto Arciniegas, Guido Ruiz, Professor Robinson

Discussed Topics: After discussion, the velocity of the team were estimated to be 30 story points.

The product owner chose the following user stories to be done during the next sprint. They are ordered based on their priority.

- User Story #284: Multiple job submission for Tools Page
- User Story #285: Redesign Tools Page (functionality)
- User Story #286: Upload Files Tool
- User Story #287: Create a log to track system usage

The team members indicated their willingness to work on the following user stories.

- Daniel Gonzalez: User Story #284, #285, #286
- Roberto Arciniegas: User Story #287

Sprint 4 Planning Report

Date: March 11th, 2016

Attendees: Daniel Gonzalez, Roberto Arciniegas, Guido Ruiz, Professor Robinson

Discussed Topics: After discussion, the velocity of the team were estimated to be 30 story points.

The product owner chose the following user stories to be done during the next sprint. They are ordered based on their priority.

- Continue working on user stories from previous sprint
- User Story #288: Profile page redesign
- User Story #289: Move pending jobs to the profile page
- User Story #290: User log-in statistics

The team members indicated their willingness to work on the following user stories.

- Daniel Gonzalez: User Story #288, #289
- Roberto Arciniegas: User Story #290

Sprint 5 Planning Report

Date: March 21st, 2016

Attendees: Daniel Gonzalez, Roberto Arciniegas, Guido Ruiz, Professor Robinson

Discussed Topics: After discussion, the velocity of the team were estimated to be 35.

The product owner chose the following user stories to be done during the next sprint. They are ordered based on their priority.

- User Story #291: Upload File Integration with Multiple Job Submission
- User Story #292: Sign-in with email
- User Story #293: Correct file names for jobs

The team members indicated their willingness to work on the following user stories.

- Daniel Gonzalez: User Story #291
- Roberto Arciniegas: User Story #292, #293

Sprint 6 Planning Report

Date: April 18th, 2016

Attendees: Guido Ruiz, Michael Robinson, Masoud Sadjadi

Discussed Topics: After discussion, the velocity of the team were estimated to be 35.

The product owner chose the following user stories to be done during the next sprint. They are ordered based on their priority.

- User Story #296: Preparing for the Final Showcase
- User Story #297: Preparing for the Final Showcase

The team members indicated their willingness to work on the following user stories.

- Daniel Gonzalez: User Story #296
- Roberto Arciniegas: User Story #297

Appendix D - Sprint Review Reports

Sprint 1 Review Report

Date: January 29th, 2016

Attendees: Daniel Gonzalez, Roberto Arciniegas, Guido Ruiz

Discussed Topics:

After a show and tell presentation, the implementation of the following user stories were accepted by the product owners:

- Our work for Sprint 0

The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Spring Planning meeting.

- None

Sprint 2 Review Report

Date: February 12th, 2016

Attendees: Daniel Gonzalez, Roberto Arciniegas, Guido Ruiz

Discussed Topics:

After a show and tell presentation, the implementation of the following user stories were accepted by the product owners:

- User Story #280
- User Story #281

The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Spring Planning meeting.

- None

Sprint 3 Review Report

Date: February 26th, 2016

Attendees: Daniel Gonzalez, Roberto Arciniegas, Guido Ruiz

Discussed Topics:

After a show and tell presentation, the implementation of the following user stories were accepted by the product owners:

- None

The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Spring Planning meeting.

- User Story #282
 - Reason for rejection:
 - User Story is incomplete. The remaining work has been pushed onto the next sprint in the form of additional user stories
 - How this should be reflected on the user story definition in Mingle:
 - New user stories have been created (#284 and #285) to reflect the missing and additional requirements
- User Story #283
 - Reason for rejection:
 - Misunderstanding over the requirements for the user story: no actual FTP server had to be set up, there just had to be an adjustment of the front-end
 - How this should be reflected on the user story definition in Mingle:
 - The user story definition should emphasize that the criteria for acceptance consist of front-end aspects of the "FTP" page, as all functionality and back-end aspects are already intact

Sprint 4 Review Report

Date: March 11th, 2016

Attendees: Daniel Gonzalez, Roberto Arciniegas, Guido Ruiz

Discussed Topics:

After a show and tell presentation, the implementation of the following user stories were accepted by the product owners:

- User Story #284
- User Story #285
- User Story #286
- User Story #287

The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Spring Planning meeting.

- None

Sprint 5 Review Report

Date: March 21st, 2016

Attendees: Daniel Gonzalez, Roberto Arciniegas, Guido Ruiz

Discussed Topics:

After a show and tell presentation, the implementation of the following user stories were accepted by the product owners:

- User Story #288
- User Story #289
- User Story #290

The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Spring Planning meeting.

- None

Sprint 6 Review Report

Date: April 18th, 2016

Attendees: Daniel Gonzalez, Roberto Arciniegas, Guido Ruiz

Discussed Topics:

After a show and tell presentation, the implementation of the following user stories were accepted by the product owners:

- User Story #291

The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Spring Planning meeting.

- User Story #292
- User Story #293

Appendix E - Sprint Retrospective Reports

Sprint 1 Retrospective

Date: January 29th, 2016

Attendees: Daniel Gonzalez, Roberto Arciniegas, Guido Ruiz

Discussed Topics:

What went wrong?

- Did we do a good job estimating our team's velocity?
 - Yes, our previous Sprint 0 went fine.
- Did we do a good job estimating the points (time required) for each user story?
 - Yes, our user stories took about as long as we had hoped.
- Did each team member work as scheduled?
 - Yes, everything was accomplished on schedule

What went right?

- Everything, our user stories for Sprint 0 were finished on time

How to address the issues in the next sprint?

- How to improve the process?
 - Continue to work as we have been
- How to improve the product?
 - Complete all user stories as requested
 - Gain the requisite familiarity with the system

Sprint 2 Retrospective

Date: February 12th, 2016

Attendees: Daniel Gonzalez, Roberto Arciniegas, Guido Ruiz

Discussed Topics:

What went wrong?

- Did we do a good job estimating our team's velocity?
 - Yes, we accomplished all of the work we intended to
- Did we do a good job estimating the points (time required) for each user story?
 - Yes, each story took about as long as we had hoped
- Did each team member work as scheduled?
 - Yes, everything was completed on time and we both cooperated appropriately

What went right?

- Again, all of our stories were completed successfully

How to address the issues in the next sprint?

- How to improve the process?
 - Continue to do our work as scheduled
- How to improve the product?
 - Continue to work on the user stories

Sprint 3 Retrospective

Date: February 26th, 2016

Attendees: Daniel Gonzalez, Roberto Arciniegas, Guido Ruiz

Discussed Topics:

What went wrong?

- Did we do a good job estimating our team's velocity?
 - No, as our story points were slightly optimistic
- Did we do a good job estimating the points (time required) for each user story?
 - No, user story #282 has taken slightly longer than initially suspected
- Did each team member work as scheduled?
 - Yes, everyone worked as scheduled

What went right?

- Significant progress on #282 was made and some of the remaining work has been integrated into User Story #284 and #285

How to address the issues in the next sprint?

- How to improve the process?
 - Must be more conservative in assigning points to user stories
- How to improve the product?
 - Continue to work on the user stories as directed

Sprint 4 Retrospective

Date: March 11th, 2016

Attendees: Daniel Gonzalez, Roberto Arciniegas, Guido Ruiz

Discussed Topics:

What went wrong?

- Did we do a good job estimating our team's velocity?
 - Yes, this time was much better
- Did we do a good job estimating the points (time required) for each user story?
 - Yes, our more conservative approach has been successful
 - However, we did not finish our user stories
- Did each team member work as scheduled?
 - Yes, each team member worked as scheduled

What went right?

- The site is looking much better now, specifically the tools page, and we are becoming more comfortable with the Agile development paradigm

How to address the issues in the next sprint?

- How to improve the process?
 - Continue to communicate with each other and monitor our productivity
- How to improve the product?
 - Continue development on the assigned user stories

Sprint 5 Retrospective

Date: March 21st, 2016

Attendees: Daniel Gonzalez, Roberto Arciniegas, Guido Ruiz

Discussed Topics:

What went wrong?

- Did we do a good job estimating our team's velocity?
 - Yes, we are now gradually increasing our team's estimated velocity to increase performance
- Did we do a good job estimating the points (time required) for each user story?
 - Yes, the user stories' points were well estimated
- Did each team member work as scheduled?
 - Yes, each team member worked as scheduled

What went right?

- We were able to deliver on all of our user cases

How to address the issues in the next sprint?

- How to improve the process?
 - Keep doing what we're doing
- How to improve the product?
 - Continue to deliver on the requested user stories and continue communication with Professor Robinson and Guido Ruiz, as some of our suggestions have become user stories for improving the software system

Sprint 6 Retrospective

Date: April 18th, 2016

Attendees: Daniel Gonzalez, Roberto Arciniegas, Guido Ruiz

Discussed Topics:

What went wrong?

- Did we do a good job estimating our team's velocity?
 - Yes, our velocity was pretty well-estimated
- Did we do a good job estimating the points (time required) for each user story?
 - Yes, the user stories' points were well estimated
- Did each team member work as scheduled?
 - Yes, each team member worked as scheduled

What went right?

- We were able to deliver on all of our user cases

How to address the issues in the next sprint?

- How to improve the process?
 - Keep doing what we're doing
- How to improve the product?
 - Effectuate any final touch-ups before the Final Showcase