

Virtual Biology Lab Simulation: Mitosis Group 11

*Steve Pak, Marvin Liu, Anthony Porturas, Brandon Lum,
Evan Chipps, Jeffrey Gillen*

[Project Website](https://sites.google.com/site/virtualbiologylab/)¹

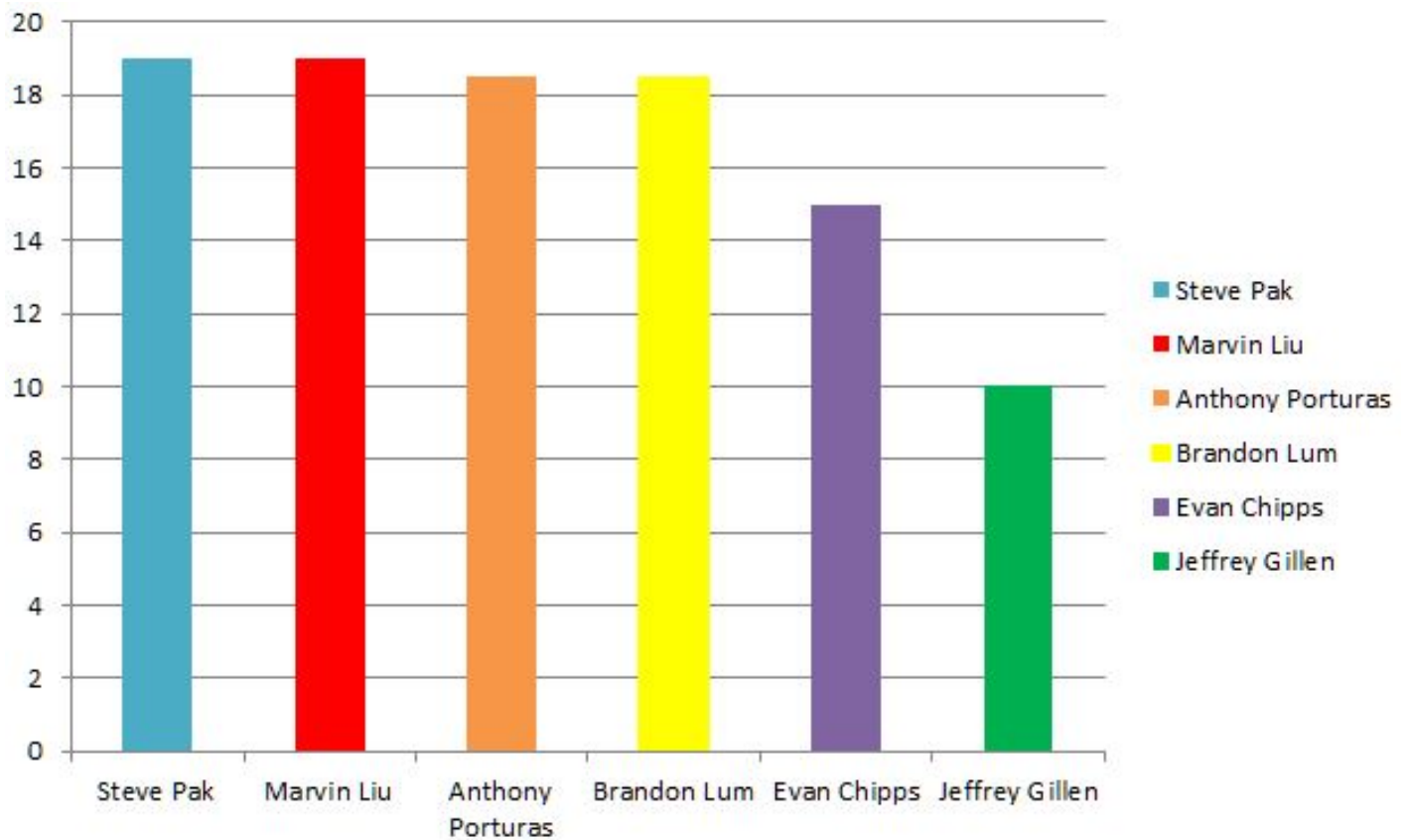
¹<https://sites.google.com/site/virtualbiologylab/>

Individual Contributions

Figure 1: Responsibility Matrix

	Steve	Marvin	Anthony	Brandon	Evan	Jeffrey
Project Management (10 pts):	19%	19%	18.5%	18.5%	15%	10%
Sec.1: Customer Statement of Requirements (9pts):	19%	19%	18.5%	18.5%	15%	10%
Sec.2: System Requirements (6pts):	19%	19%	18.5%	18.5%	15%	10%
Sec.3: Functional Requirements Specification (30pts):	19%	19%	18.5%	18.5%	15%	10%
Sec.4: User Interface Specification (15pts):	19%	19%	18.5%	18.5%	15%	10%
Sec.5: Domain Analysis (25pts)	19%	19%	18.5%	18.5%	15%	10%
Sec.6: Plan of Work (5pts)	19%	19%	18.5%	18.5%	15%	10%
Total Points (100pts)	19	19	18.5	18.5	15	10

Figure 2: Responsibility Allocation



Steve Pak 19%—19 points

Marvin Liu 19%—19 points

Anthony Porturas 18.5%—18.5 points

Brandon Lum 18.5%—18.5 points

Evan Chipps 15%—15 points

Jeff Gillen 10%—10 points

Contents

1	Customer Statement of Requirements	2
1.1	Problem Statement	2
1.2	Glossary	5
2	System Requirements	9
2.1	Functional Requirements	9
2.1.1	Acceptance Test	11
2.2	Non-Functional Requirements	12
2.2.1	Acceptance Test	13
2.3	On-Screen Appearance Requirements	14
2.3.1	Graphics	15
3	Functional Requirements Specifications	16
3.1	Stakeholders	16
3.2	Actors and Goals	17
3.3	Use Cases	18
3.3.1	Casual Description	18
3.3.2	Use Case Diagram	19
3.3.3	Traceability Matrix	20
3.3.4	Fully Dressed Use Case Description	21
3.3.5	System Sequence Diagram	24
4	User Interface Specification	26
4.1	Preliminary UI Design	26
4.2	User Effort Estimation	28
5	Domain Analysis	29
5.1	Domain Model	29
5.1.1	Concept Definitions	30
5.1.2	Association Definition	31
5.1.3	Attribute Definitions	32
5.1.4	Traceability Matrix	33
5.2	System Operation Contracts	34
6	Plan of Work	36
6.1	Project Management	36
6.2	Introduction Simulation and Navigation Portal	37
6.3	Interactive Mitosis Experiment	37
6.4	Data Collection	37
7	References	38
7.1	Biology-Related	38
7.2	Software-Related	38
7.3	Pictures	38

Note: this document will refer to students and professors as male at times, in order to simplify writing, and to avoid reading "the student" multiple times within the same sentence or paragraph. We understand that students and professors can be female as well.

1 Customer Statement of Requirements

1.1 Problem Statement

Laboratories are used every year by students worldwide to gain hands on experience simulating and observing the mysteries of science. Experiments are constructed from multiple variables, ranging from student and professor experience with materials to equipment familiarity and budget. When a new lab procedure is created, schools must acquire equipment, which can cost upwards of thousands of dollars. Depending on the schools budget, some schools may be limited to the quantity of physical devices acquired. With physical devices, maintenance becomes necessary due to the following: everyday use, age, and even misuse—which is expensive to repair or even replace. A full list of the problem domain is listed below:

1. Lab equipment is very expensive.
2. Lab equipment is limited in quantity
3. Class time is limited; as a result, the lab design may exclude some important concepts. Additionally, those unable to finish must return at another time to complete it, or risk obtaining a lower grade.
4. Some lab equipment is difficult to use without a tutorial or manual.
5. Time-consuming lab demonstrations are inexpedient.
6. Monitoring students become impossible due to teacher to student ratio, leads to a lot of wasted time, as a Lab TA can only help one group at a time, while the students that need help ended up waiting.
7. In cases of misuse, the repairing/replacing of lab equipment uses up time and money.
8. Its difficult to get an accurate lab assessment on performance in a real life experiment.
9. Interactivity is preferable for students because it engages them in learning.
10. Real life visuals may be too hard to analyze correctly for beginners.
11. Students should be able to learn at their own pace and run the lab simulation as much times as needed for better understanding..
12. Simplicity is preferred when introducing students to new subjects.
13. There is no consistent standard grading policy due to human error.
14. In regular lab assignments, students work in groups, which hinders individual learning. Some teammates may do the work while the others sit and watch.

Going to an actual laboratory can be inconvenient and cause numerous problems for both the school and students. Stress associated with these laboratory issues range from financial burdens to time constraints. Additionally, lab equipment is expensive, limited in supply, and complicated to learn for beginners. They can also become broken and misused, forcing the school to buy new equipment so other students can use them in the future.

Equipment such as microscopes, tools, and B & L Spectronic 20 spectrophotometers can cost thousands of dollars per device. The cost is significantly higher if one of each is required for each group of students in a class. Furthermore, some students will not be able to finish the lab because of a lack of supplies and equipment. They will end up waiting until another group finishes using their equipment, which wastes time. This causes stress and a flimsy knowledge of the material set as the learning objective. Due to limited time and resources, including availability of professors and Teaching Assistants, students may often receive inadequate feedback, inhibiting improvement in future labs.

To remedy these real life problems, a web-based program will be created that is capable of running the biology mitosis that is used by the General Biology (01:119:101) course at Rutgers University. A virtual biology lab provides the tools and a controlled environment for which students can perform their duties in the lab without the limitations that are generally associated with their real-world counterparts. This description of the lab experiment can be seen here:

- Lab 2: [Cell Division](#)²

Objective:.

1. Describe each stage of nuclear division (mitosis) and cytoplasmic division (cytokinesis) and identify each under the microscope
2. Use a compound microscope and describe two stains used to facilitate observation
3. Graph data correctly paying particular attention to the differences between independent and dependent variables

Virtual Biology labs solve the need for physical equipment which will greatly reduce the cost and time associated with maintaining such products. Along with such improvements, this solution will provide students with an analysis of their work as they progress through the lab in the form feedback preset by the TA and/or Professor.

The virtual biology lab would allow students to learn and execute the lab at a pace that they feel comfortable, allowing time for them to absorb and fully understand the material. Students that can grasp concepts more easily than others will no longer be hindered by those that need more time while those that need more time won't be pressured to keep up. The virtual solution will generate a more individualistic approach to the lab environment to prepare allow perform the lab to the best of their ability.

The web-based solution will consist of a simple yet aesthetically pleasing user interface consisting of some key features that make this product stand out. Some basic features included: a user interface for both students and professors, the actual lab simulation, and to observe student performance in the lab simulation, respectively. Some stand out features that are to be included with this solution: feedback preset by the TA/Professor and a strike system to provide warnings to students indicating points of error.

One of the more exclusive features, feedback, will be dynamic. Depending on the students response to an activity or question, the system will provide feedback appropriate to the selection. Example: a student selects a piece of equipment from a list, if the student selects an incorrect tool, the simulation

²<http://www.ece.rutgers.edu/~marsic/books/SE/projects/ViBE/biolab-1.pdf>

will tell the student a mistake has been made and provide the name of said equipment and ask the student to try again. The system will keep track of the mistakes and provide the professor/grade with a list for reference.

Through this website, the professors will have access to a list of features to which they can choose to include with their corresponding labs. The professor will have access to a list of features including, but not limited to: feedback system, strike system, grades, and a summary system per each student.

Following the Mitosis Lab Manual from the description above, courtesy of Rutgers University, the simulation will run as follows. At first, the simulation will start off with a learning animation, where the steps of mitosis(interphase, prophase, metaphase, anaphase, telophase) are demonstrated. This serves as a quick overview and precursor to the actual lab experiment, analogous a TA giving a brief description and lesson prior to the experiment. Following the brief learning session, the interactive portion of the experiment will begin. It will start off with the student controlling a knife, cutting a piece of the onion root, and placing it on the slide. A stain will be used by the student to make observing the cell less difficult. The student will then place the slide under a microscope, and it will show the cell, possibly out of focus, forcing the student to focus the microscope. The students will then observe the cells dividing and the different phases of Mitosis. The steps above will be repeated for the whitefish, and prompt the student to note the differences. After observing the animal cell, the simulation will then go through an interactive tutorial on graphs and table, specifically the pie chart and bar graph. Data collection is an important part of any lab experiment, and without it, results of the experiments are hard to analyze. For this reason, the team decided that data collection is a crucial part of the experiment and of the lab itself, so it is included within the simulation.

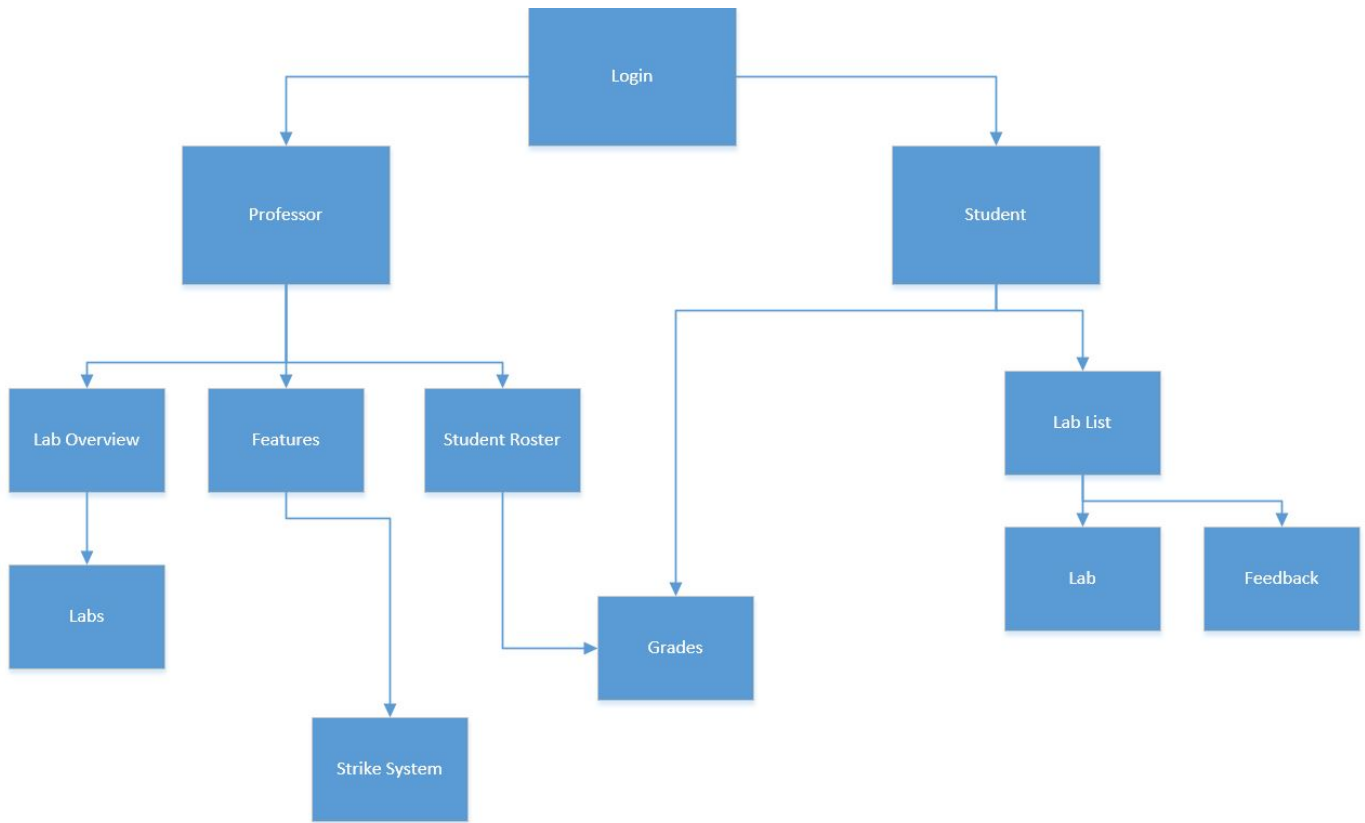
The lab manual describes its objectives below:

- describe each stage of nuclear division (mitosis) and cytoplasmic division (cytokinesis) and identify each under the microscope.
- use a compound microscope and describe two stains(acetocarmine or aceto-orcein) used to facilitate observation.
- graph data correctly, paying particular attention to the difference between independent and dependent variables.

The above simulation not only fulfills the above goals, but creates an improved solution alternative to a physical lab experiment in terms of cost and convenience.

From the virtual biology lab, users will gain the ability to perform the lab simulations from anywhere with Internet access. Since each student will be logging in using their Rutgers University netID, their grades and progress will be saved accordingly: instructors can see each students grades and mistakes. Additionally, students will be given immediate, real-time feedback if a mistake is made during the lab simulation. From this, they will be able to learn from their mistakes and better understand what changes are necessary.

Figure 3: Graphic of a diagram visualizing the functionality of the program to the different users



1.2 Glossary

Term	Definition
Anaphase	The stage of mitosis in which the chromosomes move toward the poles of the spindle.
Apical Meristem	A region of active growth (in this case, of an onion root).
Centromere	The point or region on a chromosome to which the spindle attaches during mitosis. Generally found in the center of the chromosome.
Chromatid	One of the usually paired and parallel strands of a duplicated chromosome joined by a single centromere.
Chromosome	The part of the cell that contains the genes which control
Cytokinesis	The process of in which a single cell splits itself into two equal cells through cytoplasmic division.
Cytoplasm	The organized complex of inorganic and organic substances external to the nuclear membrane of a cell.
Diploid	A cell or an organism consisting of two sets of chromosomes (2n).
Dye Solution	A solution of acetocarmine or aceto-orcein used to increase ones ability to see small structures within the cell.
Equatorial Plane	The plane perpendicular to the spindle of a dividing cell and midway between the poles.
Error	An incorrect action within the simulation, that the system detects.

Haploid	A cell or an organism consisting of one set of chromosomes (n).
Homologous	Having the same relative position, value, or structure.
Hot Plate	A machine used to heat material.
Interphase	The interval between the end of one mitotic division and the beginning of another.
Meristematic Region	The region in which cell division is occurring.
Metaphase	The stage of mitosis in which the chromosomes become arranged in the equatorial plane of the spindle.
Metaphase Plate	Also known as equatorial plane. see definition of equatorial plane.
Microscope	A device used for producing a much larger view of very small objects so that they can be clearly seen.
Microscope Slide	A thin piece of glass that is used to hold material that will be inspected under a microscope.
Mitosis	A type of nuclear division that involves a sequence of events by which the nuclear material of one cell is distributed into two equal parts.
Navigation Portal	The main menu screen that both students and professors will see as soon as they log in. This serves as a central node to access different parts of the system.
Nuclear Envelope	A double layered membrane that envelops the nucleus of a cell, separating the contents of the nucleus from the cytoplasm.
Nucleolus (Nucleoli)	A round granular structure within the nucleus of a cell, composed of protein and RNA (ribonucleic acid).
Nucleus	The central part of most cells that contains genetic material and is enclosed in a membrane.
Pole	Two structures that exist on either side of the cell that is divided by the equator.
Prophase	The initial stage of mitosis characterized by the condensation of chromosomes consisting of two chromatids, disappearance of the nucleolus and nuclear membrane, and formation of mitotic spindle.
Root Cap	A mass of cells at the end of a root (in this case, of an onion) that cover and protect the growing cells.
Stain	Also known as dye solution. see definition of dye solution.
Somatic Cell	One of the cells of the body that make up the tissues, organs, and parts of that individual other than the germ cells.
Spindle Fiber	Any of a network of filaments that collectively form a mitotic spindle that is involved in moving and segregating the chromosomes.
Telophase	The final stage of mitosis in which the spindle disappears and the nucleus reforms around each set of chromosomes.
Whitefish Blastula	An early stage in the embryonic development of a whitefish.

Table 1: Glossary of Terms

Figure 4: Cell Structure: Plant Vs Animal

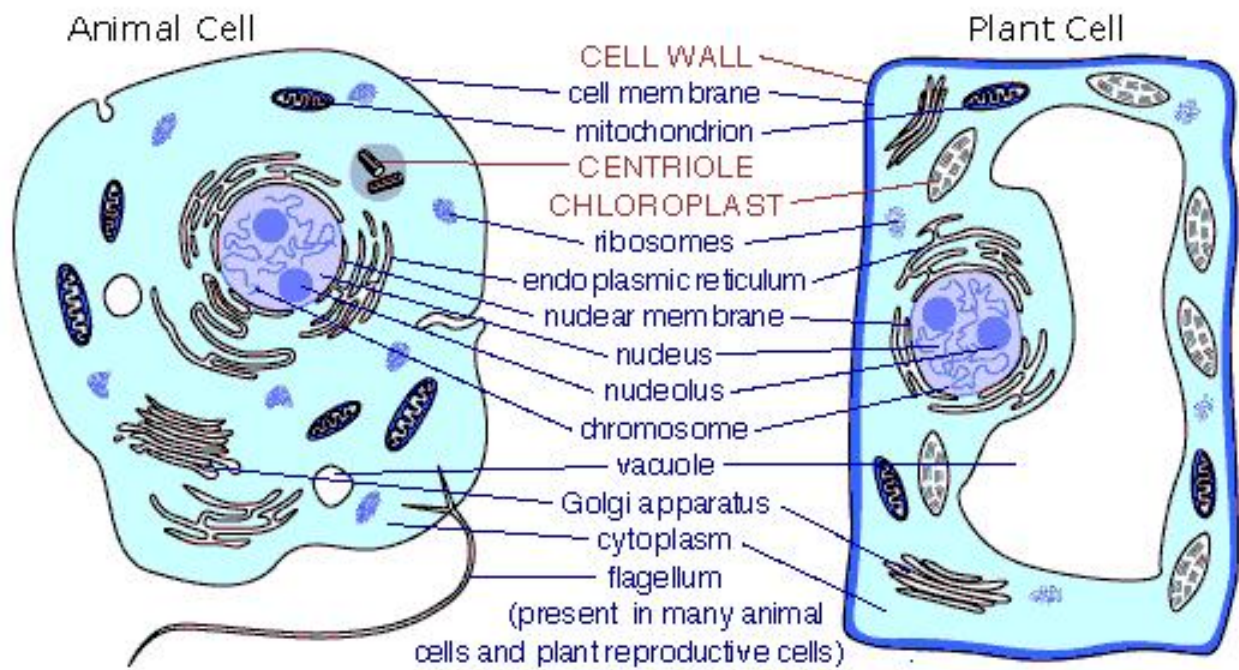
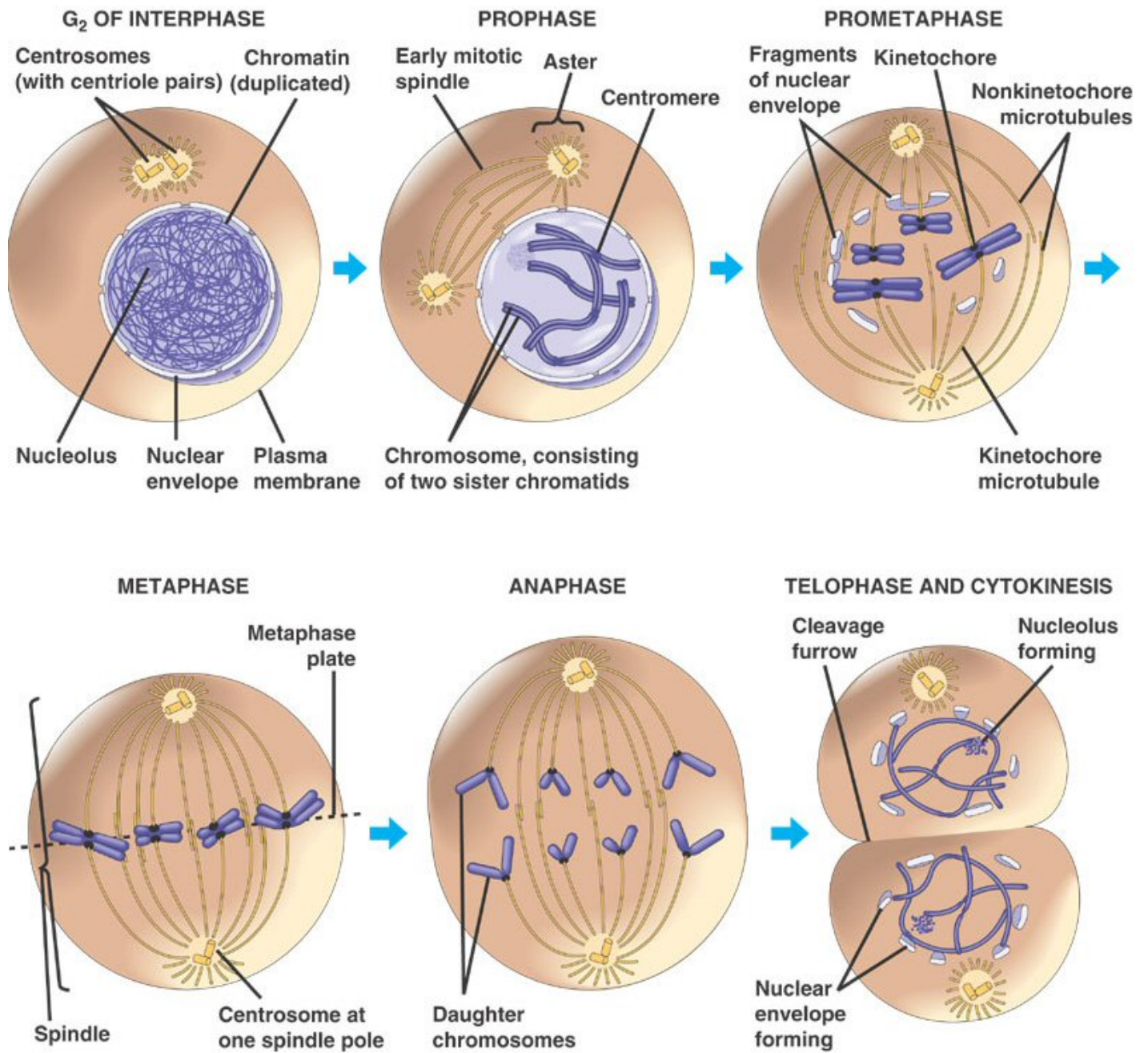


Figure 5: Stages of Mitosis



2 System Requirements

The virtual biology lab is extremely complex, as opposed to being complicated. While it seems very difficult to implement such a system, in reality, the system is essentially made up of many smaller and simpler features. The feature list of this system is both numerous and unique in nature. To simplify the need to specify which requirements are necessary and higher priority compared to other features which are optional and low priority, 2 enumerated requirement lists were created, one to specify the Functional Requirements, the other to specify the Non-Functional Requirements. Lastly, a table specifying the On-Screen Appearance Requirements is added below to explain what graphics are needed.

2.1 Functional Requirements

The team defined the functional requirements as any feature of the system that is mechanical in nature. For example, does it allow the student to run the simulation? These requirements are easy to test for, and many can be tested by simply running the program and seeing if it does what it says it does. The list is shown here: .

Identifier	Priority Weight	Requirement
REQ-1	5	The system shall run the entirety of the mitosis lab simulation when accessed by the user.
REQ-2	4	<p>The system shall allow the simulation to be interactive and complex so it has many options available. The options include:</p> <ol style="list-style-type: none">1. A wide variety of incorrect choices shall be available, along with the correct answers, to allow the students to make errors.2. The system shall allow the students to change microscope settings such as zooming in and out, to determine the best view of the slides.3. The system shall allow the user to create graphs and tables discussing the time spent in each stage of mitosis.4. The system shall allow the user to illustrate each stage of mitosis.5. The system shall have an introduction simulation slideshow to the mitosis lab.

REQ-3	4	<p>The system shall track the students actions and do the following:</p> <ol style="list-style-type: none"> 1. The system shall send a message to the student as a feedback system, showing the students answer as either correct or incorrect. 2. If the action was an error, the system will keep count of the errors, as well as record where in the simulation it occurred. The feedback system mentioned previously shall elaborate on what the student did incorrectly. 3. At the end of the simulation, a record of the student's performance shall be saved in a database.
REQ-4	3	<p>The system shall make an action when the error count reaches the amount specified by the professor:</p> <ol style="list-style-type: none"> 1. Points shall be deducted from the lab grade. 2. The student shall be forced to repeat the simulation.
REQ-5	2	The system shall allow the professor to access and edit grades associated with students that are registered for the course, accessed through the database.
REQ-6	2	The system shall allow a user to log in as professor or student.
REQ-7	2	The system shall allow new users to register for the class.
REQ-8	1	The system shall allow the student to view his or her grade, accessed through the database.

Table 2: Functional Requirements

Obviously, REQ-1 is the highest priority. The simulation is the bread and butter of this project, without it, this project has no base. The key challenges within the simulation that the team decided to address are as follows: realistic simulation of the mitosis lab, and real-time feedback on the interactions of the student. The above table exemplifies this in REQ-2 and REQ-3. The team discussed the possibilities of the simulation and decided that the lab should not be linear. Instead, there are options for the student as a user to branch out from what the lab manual deems a correct response. Another way to interact with the system, and is necessary for any lab experiment, is data collection. The system should allow the student to create graphs and tables to collect the results of the experiment. Lastly, the student should be able to use tools effectively. In this experiment, the microscope is the most widely used tool, as well as equipment relating to the microscope. The system shall have a realistic simulation of the focusing of microscopes. These features are extremely important, because, sometimes the best way to learn is to learn from your mistakes. That is why the weight of REQ-2 is high(4).

A student should have the right to know whether a mistake was made. The system lets the student know through the feedback system described in REQ-3. The system will message the student, letting them know whether they made the correct action, or if they committed an error. If the student made the correct action, they will be allowed to move on to the next step. If they committed an error, the student will have to try again until they make the correct response. The feedback system will also be responsible

for keeping track of the student errors. There is no need to keep track of the student's correct actions as the system will know the path the student took just by knowing which errors he made and which stage of the simulation he is at.

This project is heavily designed for users that are students taking biology. However, professors teaching the course should have some power and control over this system. REQ-4 and REQ-5 are directed towards the professor. The professor needs to keep track of the list of students registered for his course. Therefore, the professor should have access to a system feature that allows this, exemplified through REQ-4. The system will pull up a list of students and their ID numbers as well as grades for the lab, when the professor requests it. The professor should allow changes to be made, if need be.

A static universal grading scheme designed into the system is decidedly a bad idea. Different professors may have different policies on grading. This system will allow the professor to set options based on his policy and preference. Two over-lying options exist: Deduct points and force the student to redo the lab. A combination of the two is possible. The professor will then choose a number which represents the number of errors the student can make before a penalty ensues.

2.1.1 Acceptance Test

The acceptance test for the requirements can be seen below. The acceptance test is a way to measure the success of the project based on a specific requirement. The customer will run this test, if it passes, the requirement has been fulfilled. If the test fails, the requirement has not been fulfilled, and the project can be deemed a failure by the customer. The acceptance test cases will be labeled as ATCX.YY, where X is the requirement number and maps to REQ-X, and YY is the test case number, the first test case labeled as 1.

ATC1.01	Ensure that the computer can access the simulation. Run the simulation from start to finish, making only correct actions, to allow the simulation to move to the next step. If user can reach the end, the test is successful.
ATC2.01	Ensure that the computer can access the simulation. Run the simulation, trying to make mistakes occasionally, checking to see if there are many options available to the user. If the user has many options, and the simulation is not linear, but branches out, the test is successful.
ATC2.02	Ensure that the computer can access the simulation and is at the part of the simulation of controlling the microscope. If the zoom function works as pleases to the user, the test is successful.
ATC3.01	Ensure that the computer can access the simulation. Commit an known error in the simulation. The system should respond with some sort of feedback message. If so, the test is successful.
ATC4.01	Ensure that the computer can access the simulation and the grading policy is that the grade is reduced when the error count passes the threshold. Commit an amount of errors equal to the threshold of errors allowed. If the system responds with a message saying that the user committed too many errors, and reflect that on the grade at the end of the simulation, the test is successful.

ATC4.02	Ensure that the computer can access the simulation and the grading policy is that the simulation restarts when the error count passes the threshold. Commit an amount of errors equal to the threshold of error allowed. If the system responds with a message saying that the user committed too many errors, and restart the simulation.
---------	--

Table 3: Functional Requirements Acceptance Test Cases

These acceptance tests obviously don't go into the extreme details, but should give a good idea on how the system fulfills the requirements.

One last note: Complete analysis has not been done in the totality of the requirements, due to time constraints. Only a select important ones were chosen to demonstrate and explain how the project will function.

2.2 Non-Functional Requirements

The team defined the Non-Functional Requirements as the features that improved the quality-of-life of the system as a whole. Non-Functional Requirements usually fell under one of the category in [FURPS+](#)³, but sometimes the requirements fell into more than one category. Although Non-Functional Requirements are harder to quantify and seem trivial, they are just as important as the Functional Requirements. The list is given below:

Identifier	Priority Weight	Requirement
REQ-9	5	The lab simulation shall be accurate in showing clear differences of each stage of mitosis: interphase, prophase, metaphase, anaphase, telophase.
REQ-10	4	The feedback system shall immediately display comments on what went wrong if the students made a mistake and if they answered the questions correctly.
REQ-11	3	The system shall periodically save the students progress after every action, allowing the student to exit and resume at a later point.
REQ-12	2	The simulation shall be maneuverable, allowing for clicking next to move on and building a table of contents after each completed part of the lab experiment.
REQ-13	2	The lab simulation shall be accurate in showing clear differences between plant and animal cells.
REQ-14	2	The system should include a navigation portal which serves as a central node to access other menus, following after the log in.

Table 4: Non-Functional Requirements

As state above before the table, Non-functional requirements are more about the quality of the content as opposed to the content itself. REQ-9 is a necessary requirement. This project is to be used as a learning tool, and if students do not learn the proper differences of each stage of this lab, this

³<http://en.wikipedia.org/wiki/FURPS>

project would be a failure. Each stage should be clearly defined and should not be confusing in any way. For this reason, REQ-8 is labeled with priority weight of 5. A similar requirement, REQ-13, is also essential to teach the students, although, it is a secondary goal of the lab, not a main goal

In that same vein, REQ-10 allows the student to learn from their mistake. This simulation should not be seen as a cruel test of their knowledge, but a tool to learn new things, and to reinforce what students already know. It is not meant to find each and every way to make a student fail. That is why the feedback system should be helpful and encouraging and allow the student to know exactly where they committed an error. This project as a whole should make biology more enjoyable and easier to learn. Another benefit is that students rely less on a human TA who can only do so much. For example, a class of 25 students are performing an experiment. During the experiment, students may have to ask the lab instructor questions. In cases where multiple student have a question, there is a lot of wasted time, as the instructor must go to each student with a question. This problem would be solved if this requirement is fulfilled, which is why it has a weight of 4.

Sometimes, students cannot finish the lab in the allotted time. In the case of physical lab experiments, the student would have to clean up his station, and then reset his station when he can work on it again. In the virtual lab, this problem doesn't exist. There is no need for a station set up, or clean up. If the student doesn't finish on time, the system will save his progress, and allow him to come back whenever he's available. The way this works is similar to a save point. These save points would be set up every one or two steps in the simulation, and if the student passes the step, the system will save that state. That way, if the student leaves in the next step, when he comes back, the system will load up that state. In that same sense, the student will also backtrack his progress if he wants to reinforce the concepts. These features can be seen in REQ-11 and REQ-12.

The system needs to be able to navigate through the different menus. This can be done through the Navigation Portal. This is set at a low priority as it is not necessary for the system to be successful, set at a weight of 2.

2.2.1 Acceptance Test

Following the conventions in the functional requirement acceptance tests, here are the nonfunctional requirement acceptance tests.

ATC9.01	Ensure that the computer has access to the simulation. Navigate through the simulation, making sure that the different stages are well defined and correct, following the definitions in the glossary. If so, the test is successful.
ATC10.01	Ensure that the computer has access to the simulation. Commit an known error in the simulation. If the feedback message is unhelpful, or absent this test fails. If the message is very clear and explains the mistake well, the test is successful.
ATC11.01	Ensure that the computer has access to the simulation. Start the simulation, and stop somewhere in the middle, and exit the simulation. Access the simulation again, and if the simulation starts at a near point previous to the point where the user left off, with the progress saved, the test is successful.

Table 5: Non-Functional Requirements Acceptance Test Cases

Once again, this is not an exhaustive list of the acceptance test cases for all of the requirements. It

does, however give the customer a very good feel on the fulfilment of the higher priority requirements. A final note: All of the test have only 2 final state, success or failure. If the test says that a condition makes the test successful, and that condition isn't met, the test is considered failed.

2.3 On-Screen Appearance Requirements

In order to have a simulation, graphic images of each object in the lab must be found or created. Since it is hard to determine whether a slide or a knife has more priority, the team decided that most trivial, internet searchable image had the same priority, but other graphics that are specific to this system, and must be created from scratch have higher priority. For example, the Navigation Portal background is specific to the system where as a knife can be very simply found by doing a quick google search.

Identifier	Priority Weight	Requirement
REQ-14	5	<p>The system shall have an aesthetically pleasing user interface. This interface will be capable of displaying the following things:</p> <ol style="list-style-type: none"> 1. The log in screen for both student and professor. 2. The Navigation Portal, with menu buttons to access the simulation or grades for any user, or the grading scheme menu and the manage users menu for professors. 3. The simulation itself which includes the graphics for each physical element of the lab and the different pictures of mitosis and the cell as well as the tools. 4. Feedback messages to the screen sent by the feedback system. 5. A registration screen for non-users.

Table 6: On-Screen Appearance Requirements

The On-Screen Appearance requirements are essential to this project and system. Without any of these items, the simulation cannot run. If the students cannot see the different stages of mitosis, or if a graphic of a microscope was missing, the system would not seem to work, even if it was coded properly. For this reason, this requirement was given a weight of 5.

2.3.1 Graphics

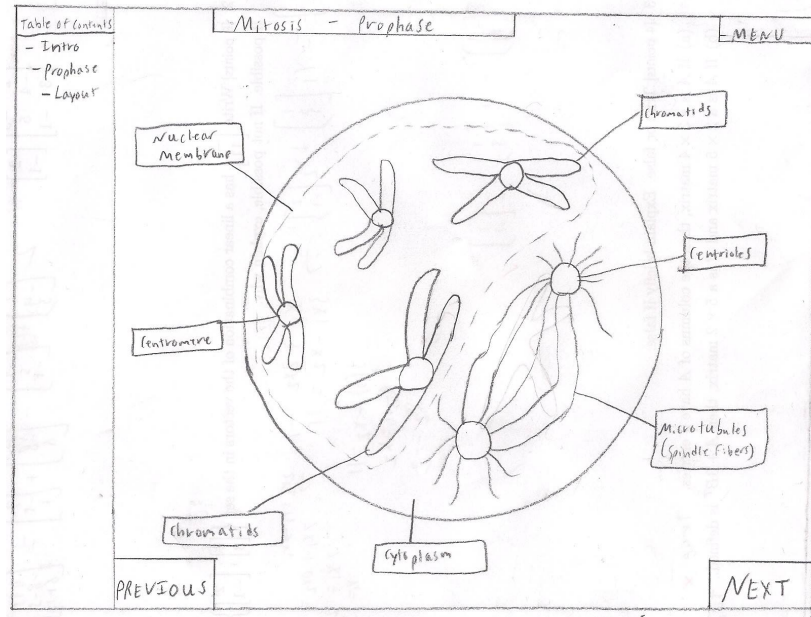


Figure 6: User Interface

3 Functional Requirements Specifications

This section contains information about the several use cases, describing different scenarios that a user may go through, stakeholders of this project will be described, a discription of the actors and goals, and some system sequence diagrams on the higher priority use cases.

Although the problem statement specifically stated that this would be a web based project, there was no mention of a database. However, a database is necessary for some of the features of the program, including: Professor has access to student performance(grades), Students are allowed to continue the experiment where they left off. The purpose and the responsibility of the database is to store data, that is accessible to the user, based on these use cases, which determines permissions of each user. First, we discuss the stake holders.

3.1 Stakeholders

A stakeholder is described as anyone or any company/organization that may have interest in the project. A list of potential stakeholders, and their reason of interest is listed below:

High Schools/Universities

The reason of interest is at first glance very obvious. It costs a lost of money to build and maintain a lab. With all the costs that a University goes through, it is very obvious that they would choose to opt out of buying expensive equipment handled by inexperienced students, and choose to go with a cheaper Virtual Biology Lab that takes minial maintenance.

Professors/Teachers

Professors and Teachers are extremely busy and have their own agendas. Sometimes they don't want, or even have time, to deal with issues with certain lab experiments. There is so much room for error and it is impossible to fix all the flaws of a physical lab.

Students

Students would have a great interest in this product. The virtual solution will be more intuitive and learning based, so that students spend more time learning instead of sitting and waiting for a professor. Because of this, students may finish labs faster and spend time studying for other classes.

Lab Technicians

Dealing with broken machinery and replacing tools that mischievous students decided to ruin creates a lot of unnecessary work for the lab technicians. Instead, with this new virtual solution, they would be able to simply maintain a system, and install software.

Not all potential users are included in this stakeholders list, as this project may find use outside of its intended purpose. However, these four entities comprise of the main audience that this project is targeting.

3.2 Actors and Goals

An actor is defined as any person or device that interacts with the system. An initiating actor can initiate a use case of the system while a participating actor cannot initiate a use case, but will still interact with the system. A list of actors and their goals are listed.

Initiating Actors

- Professor
 - to manage student users(add/remove) as well as view/edit grades
 - to create laboratory grading policies
 - to view student progress
 - to log in
- Student
 - to perform lab simulation experiment and receive feedback
 - to view grades
 - to track and save his/her progress, and backtrack
 - to log in
- Unregistered User
 - to register for lab course

Participating Actors

- Feedback system
- Database

The 2 main human users that will actively use this system are the professor and the student. Since the system is reactionary to the users actions, and should not initiate any situations, the Professor and the User will be considered as initiating actors. One more initiating actor is the Unregistered User. These are the users that the system does not recognize, because they never registered into the system. The unregistered user will eventually fall into the other 2 initiating actors, if they intend to access the system.

This system is built around the goals of the initiating actors. For this reason, a lot, if not all of the goals of the students are included as features of the system. An example of this is as follows: The system is built around lessening the load on the Professor. By allowing them to create grading policies, professors no longer have a need to observe each and every lab group to judge performance. The listed goals are a result of the problem statement and analysis.

Participating actors are entities that cannot initiate any use cases, but will still respond and interact to use cases. The database and the feedback system, the 2 participating actors in this system, directly/indirectly interact with the users and make a reaction/response to a certain action. Because they have individual responsibilities that the system rely on, but cannot make an action until a initiating actor makes an action, they are considered participating actors.

3.3 Use Cases

Use cases are essential in mapping out the capabilities of any system. The Virtual Biology Lab consists of 9 use cases, and derive from the requirements(see requirements section). One thing to note, use cases are any situation the user can use the system. One thing to note, it can only be considered a use case if an actor initiates a use case through an action. Waiting for an action does not qualify as a use case in itself. The use cases below show the various ways the user can utilize the system. A use case diagram, which really captures the interactions, can also be seen below.

3.3.1 Casual Description

UC1: Navigation Portal

A user will be able to access grades, the simulation, or pick the grading scheme. Depending on which user is logged in will determine which options will be chosen. For example, if the professor is logged in, the navigation portal will allow him to pick the grading scheme and see the grades of the students that are registered for the lab. This will fulfill the requirements below.

Derived from REQ-1, REQ-5, REQ-8, and REQ-14.

UC2: Log In

A user can log in from the main page. The user will input their username and password to which the system will recognize the portal to which it should send the user based on the information provided at registration (mandatory sub use case, <<include>> from UC-1: NavigationPortal).

Derived from REQ-6.

UC3: Register

A non-registered user will register with the system, providing user name (usually NetID), password, and type of individual (student or professor) with valid verification.

Derived from REQ-7.

UC4: ViewGrades

The students will be able to view their own grades. The professor will be able to view the grades of each student and change them accordingly (optional sub use case, <<extend>>UC-1: NavigationPortal).

Derived from REQ-5 and REQ-8

UC5: SelectGradingScheme

The professor will be able to choose how to grade the lab. He will be able to choose how many strikes are allowed, what to do when a student makes an error, etc. (optional sub use case, <<extend>>UC-1: NavigationPortal).

Derived from REQ-4.

UC6: ManageUsers

Students that sign up for the course will become alphabetized. The administer of the course (in this case, the professor) will be able to add and remove students from the database so the virtual lab can be reused. This will ensure the flexibility of the lab (optional sub use case, <<extend>>UC-4: ViewGrades).

Derived from REQ-5.

UC7: GiveFeedback

The system will respond with messages depending on the action of the user. It will give specific feedback that corresponds to a correct or incorrect answer such as acknowledgement of the correct answer, an explanation, or tips to get to the correct answer (mandatory sub use case, <<include>>from UC-8: Simulation).

Derived from REQ-2 and REQ-10.

UC8: Simulation

A user can watch and participate in the mitosis lab simulation. They will watch the animation on the screen, read relative text and interact with the virtual lab when prompted to do so. This interaction between the user and what is happening on-screen is what makes the virtual bio lab functional. (optional sub use case, <<extend>>UC-1: NavigationPortal).

Derived from REQ-1, REQ-2, REQ-9, REQ-12, and REQ-13.

UC9: ManageHistory

The system will record the progress and each step that the students make. The students will be able to exit at any time and enter where they left off (mandatory sub use case, <<include>>from UC-8: Simulation).

Derived from REQ-3 and REQ-11.

One thing to keep in mind, is that any user is not directly interacting with another user. However, for the system to use all the use cases, the bare minimum of users is 1 professor and 1 student. The casual description shows what each user is capable of, and what actions can be made while also showing how the system reacts and responds to those actions. Every use case derives from at least 1 requirement.

3.3.2 Use Case Diagram

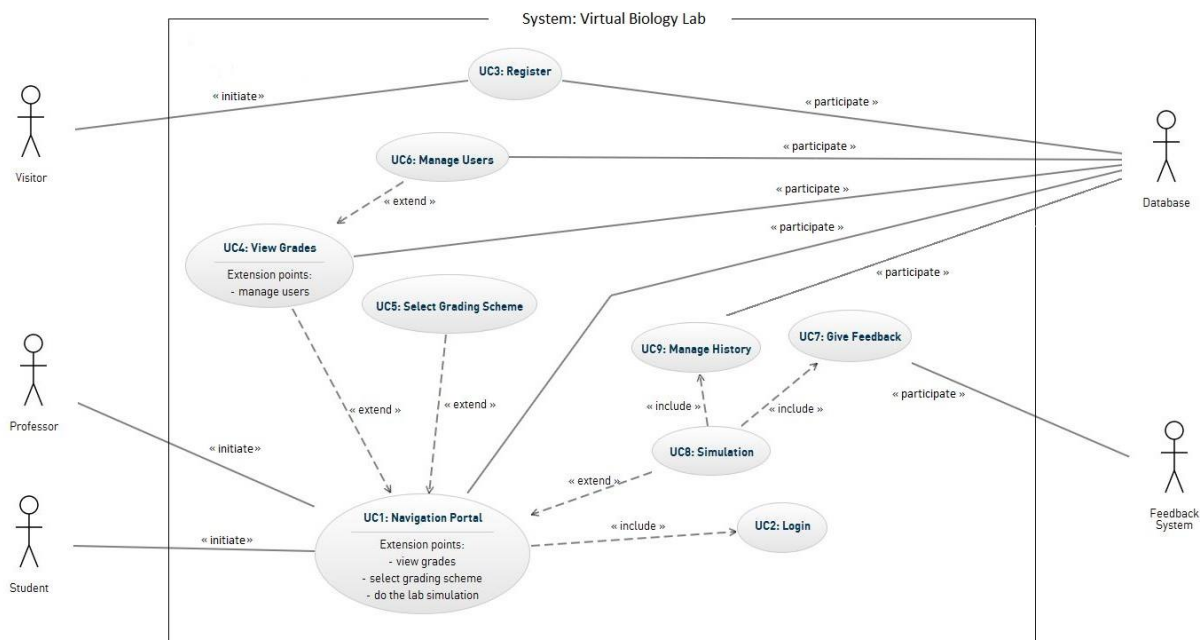


Figure 7: Use Case Diagram of Full System

3.3.3 Traceability Matrix

REQ-#	PW	UC1	UC2	UC3	UC4	UC5	UC6	UC7	UC8	UC9
REQ-1	5	X							X	
REQ-2	4							X	X	
REQ-3	4									X
REQ-4	3					X				
REQ-5	2	X			X		X			
REQ-6	2		X							
REQ-7	2			X						
REQ-8	1	X			X					
REQ-9	5								X	
REQ-10	4							X		
REQ-11	3									X
REQ-12	2								X	
REQ-13	2								X	
REQ-14	2	X								

Table 7: Traceability Matrix

The traceability matrix shows how the system requirements that we came up map to our use cases. We calculated the priority weights of the use cases, and therefore, we can order our use cases by priority:

$UC8 > UC1, UC7 > UC9 > UC4, UC5 > UC2, UC3, UC6$

It makes absolute sense that the realistic laboratory simulation and the feedback system are highest priority, as they are our most distinguishable and unique features that the system is based around. The navigation portal is also a high priority use case, because a lot of the use cases rely on the navigation portal, including the login as well as accessing the settings.

3.3.4 Fully Dressed Use Case Description

We selected UC8, UC7, UC1 to detail in the fully dressed use case description. The fully-dressed description explain the purpose of the use case, the role of each of the actors involved, what qualifies as a successful as well as a failed end condition, and the flow of events. It also explains how the system reaches each end condition through the flow of events.

Use Case UC8:	Simulation
Related Requirements:	REQ-1, REQ-2, REQ-5, REQ-12, REQ-13
Initiating Actor:	Student
Actor's Goal:	To view and interact with a simulation of a mitosis lab
Participating Actors:	Professor, Database, Feedback System
Preconditions:	Student is a registered user
Success End Condition:	The simulation is completed and the grade is recorded and saved in the Database.
Failed End Condition:	The simulation is not finished and/or progress is not saved in the Database.
Flow of Events for Main Success Scenario:	
include::GiveFeedback (UC-7), include::ManageHistory (UC-9)	
– >	1. Student accesses the simulation through the navigation portal
< –	2. System displays an introduction to the mitosis lab.
– >	3. Student interacts with the simulation by dragging and dropping, typing, or clicking.
< –	4. System gives feedback to the student.
< –	5. System will create a table of contents for each part that is correctly completed by the student.
	Step 3, 4, and 5 are repeated until the simulation is over.
< –	6. System gives a grade to the student and the corresponding grade is recorded in the Database.
Flow of Events for Extensions (Alternate Scenarios):	
	6a. System does not have a grade to input into the Database.
< –	1. System automatically gives a grade of a 0 and is recorded in the Database.

Table 8: Fully-Dressed Description, UC8: Simulation

The determined success condition is that the simulation is completed by the student. This is considered to be a success because the student's goal is to interact with the simulation of the lab, and the student is trying to learn more about mitosis and complete the lab for a grade. To complete the lab is to fulfill the goal of the student. If the student were to not finish, the success end condition has not been met, so it is considered a failure, until he completes the simulation. Even worse, if, for some reason the progress didn't save and get stored in the database, it would be as if the student did not perform anything. No grade would be recorded. This would be considered a true failure.

In order to get into the simulation, the student would have to log in, in order for the system to recognize what kind of user is accessing the system. This use case contains this as a precondition. The simulation option is available through the navigation portal, so the student should click the option for the simulation from the portal. The student will then interact with the simulation that the system will provide. As the student progresses through the simulation, the database will store the state of the student and a table of contents of his history will appear(see UI specification section). This continues, while the system is giving feedback and tracking errors. When the simulation ends, the student will be sent to a conclusion page which shows his grade based on his errors.

Use Case UC7:	Give Feedback
Related Requirements:	REQ-2, REQ-10
Initiating Actor:	Student
Actor's Goal:	To receive feedback based on given answer
Participating Actors:	Feedback System, Database
Preconditions:	The student is logged in and starts and interacts with the simulation.
Success End Condition:	Feedback is received by the student and will know what to do.
Failed End Condition:	No Feedback is given
Flow of Events for Main Success Scenario:	
include::GiveFeedback (UC-7), include::ManageHistory (UC-9)	
– >	1. Student interacts with the simulation
< –	2. System gives feedback messages based on the answer that the student gives.
– >	3. Student moves onto the next step.
	Step 1, 2, and 3 are repeated until the simulation is over.
Flow of Events for Extensions (Alternate Scenarios):	
< –	2a. Student makes an incorrect answer. 1. System will give tips on how to improve the students answer and add 1 to the number of strikes that are allowed. The Database will keep track of how many strikes there are.
< –	2. System will return back to the screen that the student made an error in so that he/she can fix the mistake.

Table 9: Fully-Dressed Description, UC1: Simulation

The feedback system is arguably the second most important feature of this system, after the simulation. Students want to know how they are performing, while the professor does not want to constantly watch over every group. UC-7 starts off with a student interaction of the feedback system as a precondition. Based on that interaction, the feedback system will then send a message to the user, and send the student to the next step. This process will then repeat until the simulation is over. If no feedback is given, based on the student action, the student cannot tell if they committed an error. This is considered a failure, because the goal was not met.

The alternative scenario of the student committing an error flow of events is as follows: the student makes an error. The system will then send a message giving tips and telling the student that his action was not the correct action to take. The system will then increment the error count(strikes). The system will then return the user to the previous stage, allowing him to try again, this time with helpful tips to allow him to continue.

Use Case UC1:	Navigation Portal
Related Requirements:	REQ-1, REQ-5, REQ-8, REQ-14
Initiating Actor:	Student, Professor
Actor's Goal:	To choose what they want to view.
Participating Actors:	Database
Preconditions:	Student and Professor are registered users.
Success End Condition:	The user successfully ends up where they wants to go.
Failed End Condition:	The user ends up at the incorrect location.
Flow of Events for Main Success Scenario:	
include::Login (UC-2)	
– >	1. The system authenticates whether or not the user is a student or a professor and decide which portal to go to.
< –	2. The system displays all available options for the user to navigate through.
– >	3. User selects which window to open and view.
< –	4. The system responds by displaying the selected option.
< –	5. System will create a table of contents for each part that is correctly completed by the student.
Flow of Events for Extensions (Alternate Scenarios):	
	3a. The system transfers to the incorrect webpage.
– >	1. The user selects the menu button and selects the navigation portal.
< –	2. The system displays all available options for the user to navigate through.
– >	3. The user selects which window to open and view.
< –	4. The system responds by displaying the selected option.

Table 10: Fully-Dressed Description, UC1: Navigation Portal

Users will be able to navigate to their desired web page using the navigation portal. A users login credentials will be validated by the system before proceeding to the appropriate navigation portal. The navigation portal will display all of the possible navigation choices to the user after they have successfully logged in. Once a choice is selected the navigation portal will bring the user to the selected location. If the user wishes to change their selection, they will be able to navigate back to the navigation portal and select another location.

3.3.5 System Sequence Diagram

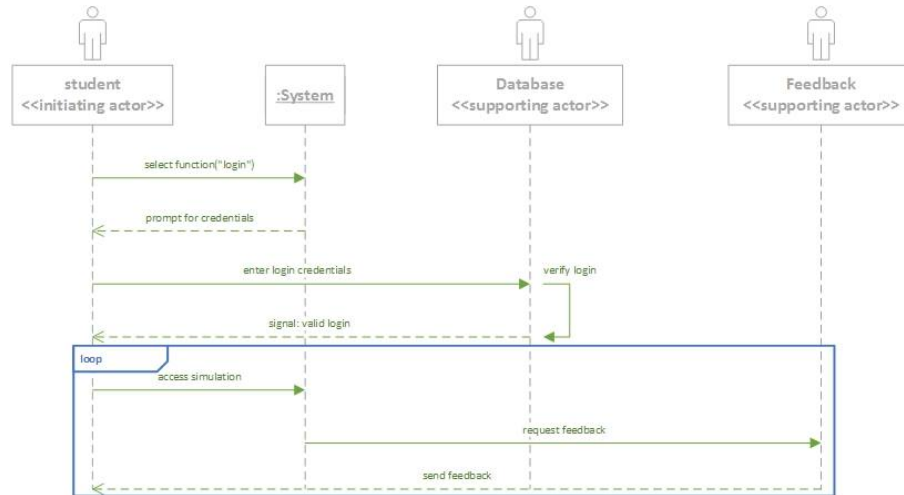


Figure 8: System Sequence Diagram of UC8: Simulation

The professor and the student are two initiating actors who can both communicate with the system in different ways. These two use cases were chosen with the intention of illustrating the capabilities of the entire system in interacting with both different actors. In the first use case, the student is the initiating actor of the sequence diagram. Initially the user calls the login function. This function causes the system to prompt the user for their credentials. Upon entering their login credentials, the input information will be checked against the stored information in the database. The database then confirms the user is a student, and it signals to the user that the credentials are verified. The student will be allowed to access the lab simulation. While performing the lab simulations, the system will periodically communicate to feedback, which is a supporting actor. When the system communicates with feedback, feedback will send information to the student concerning performance on the simulation.

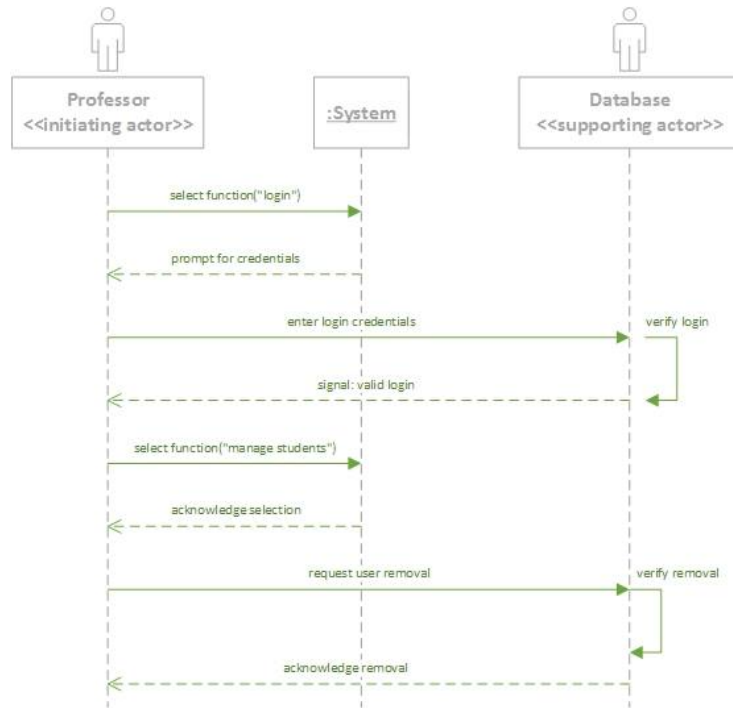


Figure 9: System Sequence Diagram of UC6: Manage Users

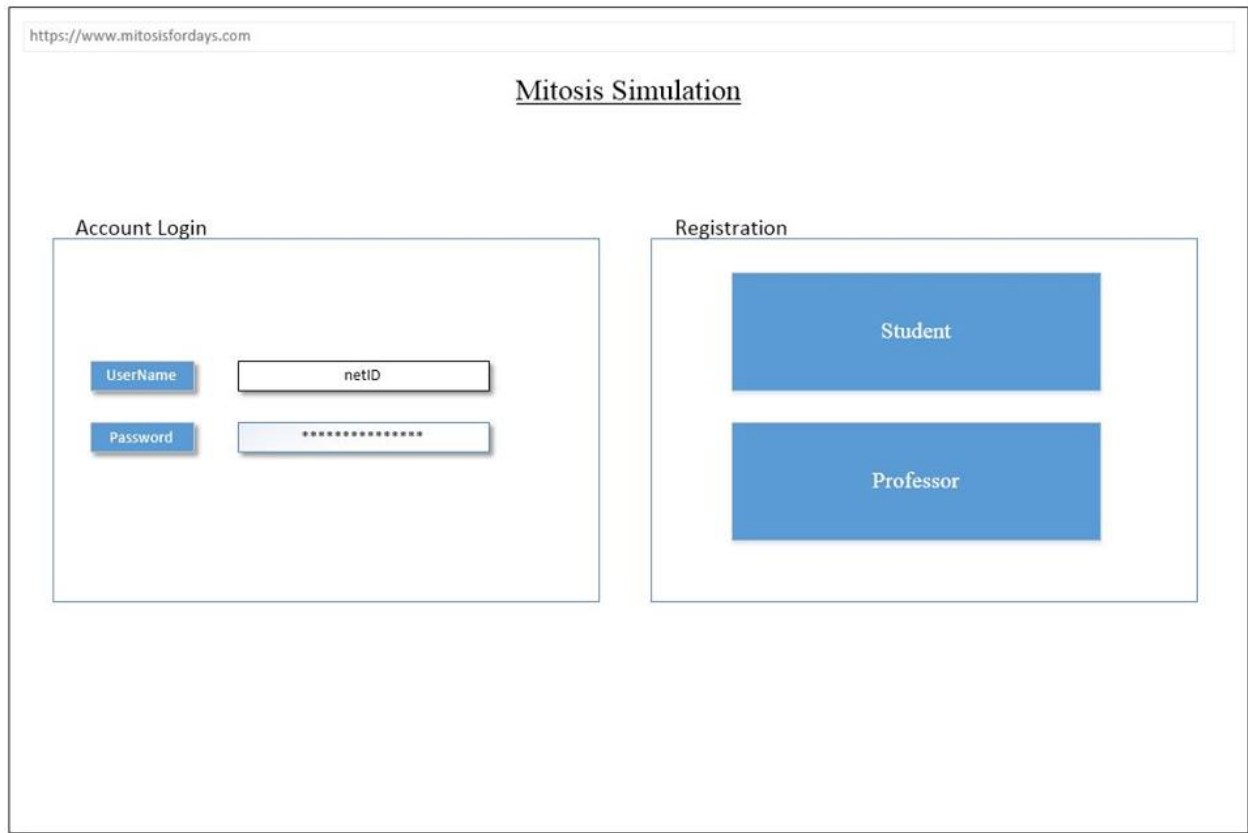
The next use case involves the professor communicating with the system. The professor has a similar foundation to its sequence diagram. The login process is identical, where the system prompts the user for their credentials. Upon the professor entering the proper credentials, the information is cross referenced against information stored in the database. Once it is confirmed that the user is a professor, the database will signal to the user that the login was valid. The professor then selects the manage students function from the system. Once the system acknowledges this call, the professor requests the database to remove a user. The database will then recognize the fact that a verifiable user has made this request, and will signal to the user that the removal has been made.

4 User Interface Specification

The above sketch (Figure 6) shows a crude expectation of the on-screen appearance of the User Interface (UI). Now that the use cases have been defined and detailed, preliminary UI designs can be created. They are considered preliminary because they are subject to change based on the progress.

4.1 Preliminary UI Design

The team decided to use the highest priority use case to display the preliminary UI.



The image shows a web browser window with the address bar displaying `https://www.mitosisfordays.com`. The main heading is Mitosis Simulation. Below the heading, there are two distinct sections: 'Account Login' on the left and 'Registration' on the right. The 'Account Login' section contains two blue buttons labeled 'UserName' and 'Password' on the left, and two white input fields on the right. The first input field is labeled 'netID' and the second is filled with ten asterisks. The 'Registration' section contains two large blue rectangular buttons. The top button is labeled 'Student' and the bottom button is labeled 'Professor'.

Figure 10: Preliminary UI Design

The Login Page is used for users to login to access their accounts as well as unregistered users to create a new account for use on site. The user will use the left info box to login if they already have an account created. The right info box is meant for those that are looking to register. The first link labelled Student in the registration box is for student registration while the second link labelled Professor is used for professors to register. Each link is customized for the each type of user.

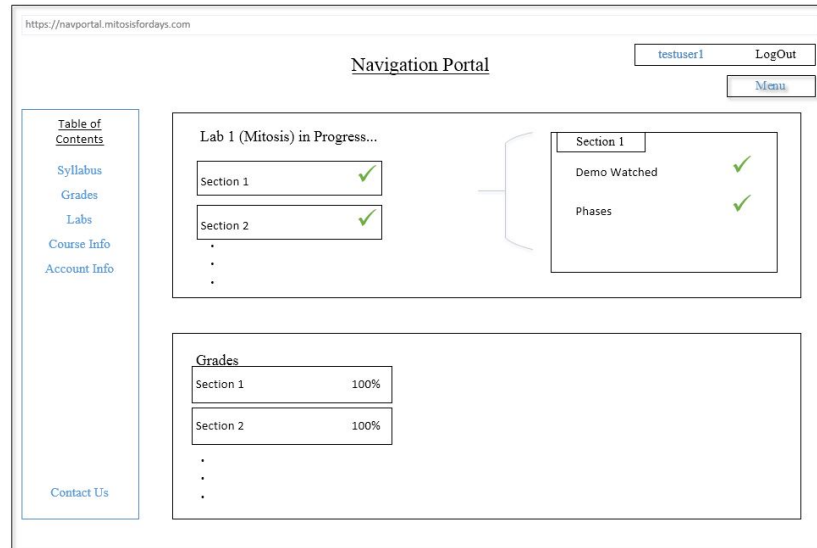


Figure 11: Preliminary UI Design

The navigation portal is a central hub for students and professors to access their labs, grades, accounts, and the simulation. As one can see, the left side of the site has a navigation Table of Contents which allows users to jump to different portals throughout the site with just one click. Here at the navigation portal, the main modules of the page can be customizable per each persons preference. In the current mockup, one can see this users set the current lab as their first module and the second module as Grades.

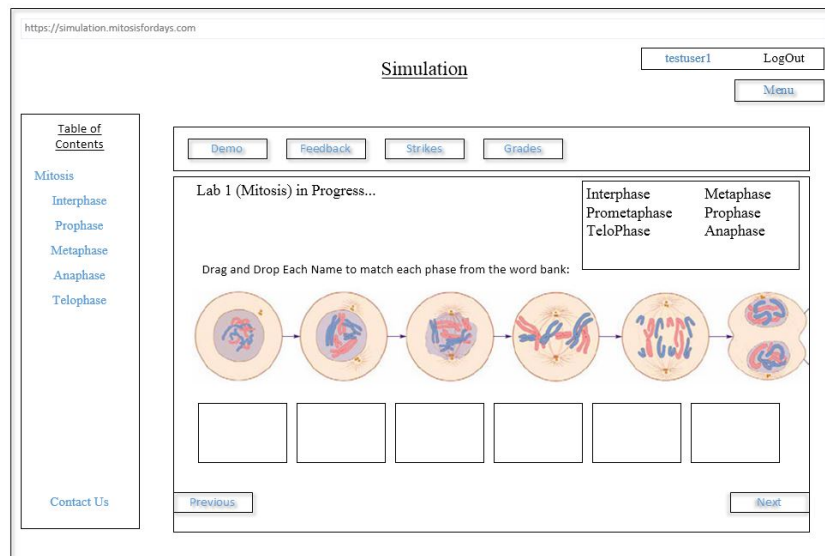


Figure 12: Preliminary UI Design

The simulation portal hosts the simulation for the current lab selected. The current one on the mockup shows the Mitosis lab in progress and an example of what a part of the simulation may contain. Above the simulation box, there is another context box which contains links associated with the current lab such as demos, feedback, strikes, and grades.

4.2 User Effort Estimation

Below is a calculation of the amount of mouse clicks and key strokes to navigate to the simulation, UC8.

1. Navigation: total 5 mouse clicks
 - (a) Click "UserName" —Enter information in Data Entry for UserName—
 - (b) Click "Password" —Enter information shown in Data Entry for Password—
 - (c) Hit "Enter" to log in
 - (d) Click "Labs"
 - (e) Click Lab 1
 - (f) Click Simulation
2. Data Entry: total 0 mouse clicks and 22 keystrokes
 - (a) Type testuser1
 - (b) Type *****

5 Domain Analysis

5.1 Domain Model

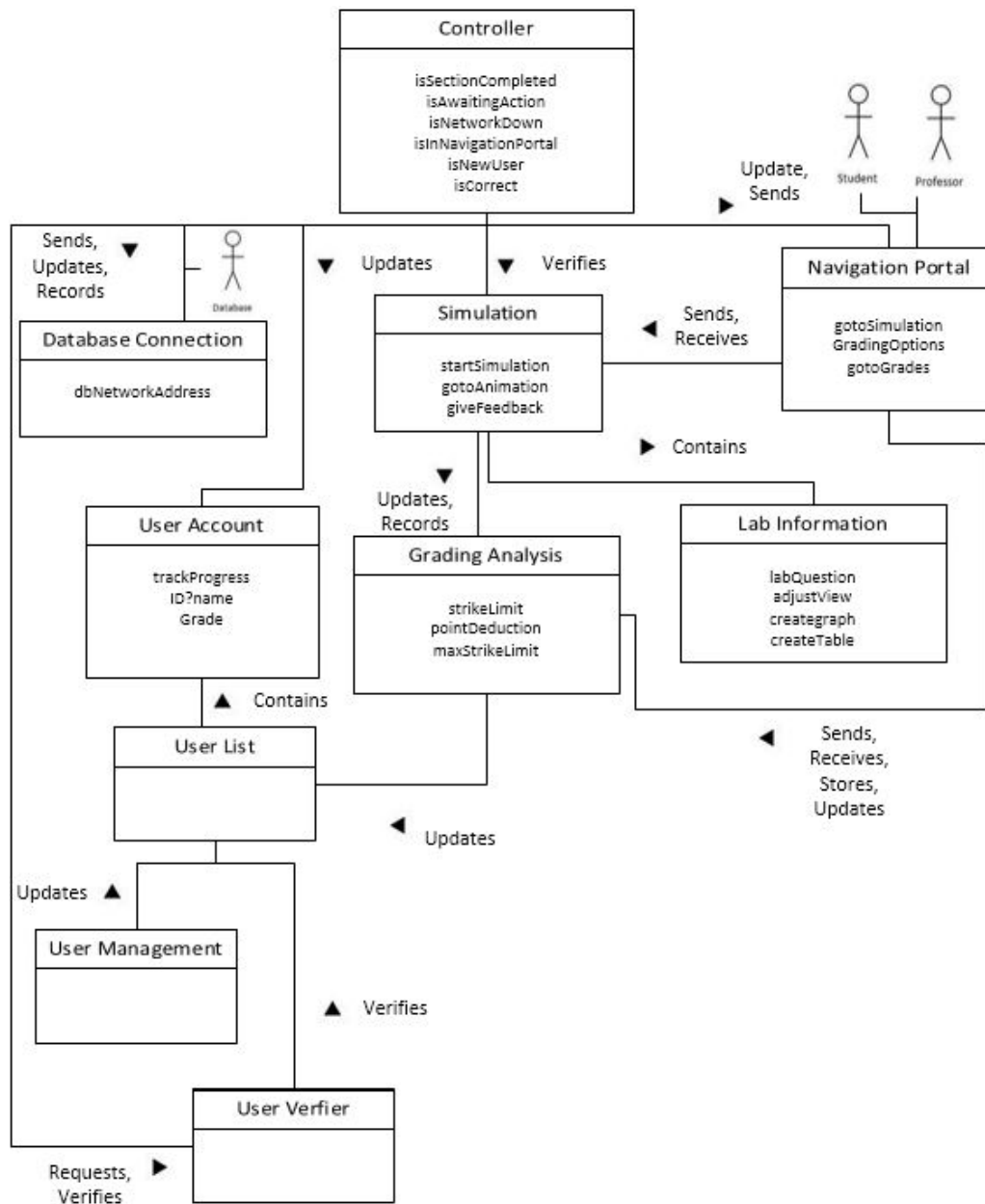


Figure 13: UML Domain Model

5.1.1 Concept Definitions

R#	Responsibility Description	Type	Concept Name
R1	Allows the user, depending on whether it is a student or a professor, to browse through the many different options given such as accessing grades, the simulation or pick the grading scheme.	D	NavigationPortal
R2	Manage interactions with the database		Database Connection
R3	Coordinate activity and delegate work originating from the User in a way that follows the use case diagram.	D	Controller
R4	Prompts the user to take action onto the lab simulation.	D	Controller
R5	Prompts the user to move onto the next or previous step.	D	Controller
R6	Verifies if the user is a registered student or professor for a specific course.	D	User Verifier
R7	Registers a new user as a professor or student into the database.	D	User Management
R8	Allows professor to insert a student into the lab course by using his NetID.	D	User Management
R9	The list of all registered students, TAs and Professors in the lab and associated grades	K	User List
R10	Keeps track of the points which also includes points taken off based on the number of mistakes made during the lab.	D	Grading Analysis
R11	Allows the professor to choose how he wants the lab to be graded.	D	Grading Analysis
R12	Allows the student to do the lab simulation.	D	Simulation
R13	Displays an animation to introduce and demonstrate the lab material to student.		Simulation
R14	Holds Lab information and data gathered by the student	K	Lab Information
R15	Keeps status of the local user and his/her progress	K	User Account
R16	Provides a useful tip or states that an answer is correct when the student makes an action.	K	Simulation

Figure 14: Concept Definitions

The concept definitions can be derived from the different responsibilities, which came from the use cases of the system. Our concepts include: Navigation Portal, Database Connection, Controller, User Verifier, User Management, User List, Grading Analysis, Simulation, Lab Information, and User Account. The responsibilities of each concept is listed above. The Navigation Portal is responsible for allowing the user to access the different options, such as getting to the grades page or to go to the simulation.

The Database connection is responsible for getting the data from other concepts into the database. The responsibility table should reflect that. The controller helps the user interact with the system, acting somewhat like a game pad in the fact that the user inputs something to the controller which will then interact with the inner system, and then the controller will tell the user to make an action prompt. User verifier has the sole task of checking whether the user is a professor or a student.

User management concept allows the system to give users(Professors) to change the User list, to add or remove or edit, and the user list is a list of all the students and their associated grades as well as the professors. The grading analysis is responsible for the grading scheme as well as keeping track of the grades and updating the user list grades. The simulation concept is responsible for the access of the actual simulation as well as the feedback messages. The User account holds each users information(Name, Type of user, grade), and the User list is comprised of all these User Accounts. The lab information holds the information on the experiment and sends data to the simulation.

5.1.2 Association Definition

Concept Pair	Association Descriptions	Association Name
Controller ↔ Simulation	Controller verifies whether or not the answers in the Simulation are correct	verifies
User Verifier ↔ User List	User Verifier verifies if the user is registered by comparing with the User List	verifies
Controller ↔ Navigation Portal	Controller updates which portal to send the individual from the Navigation Portal	updates, sends
Navigation Portal ↔ Simulation	Navigation Portal sends the user access to the Simulation and lets the user receive the lab simulation	sends, recieves
Navigation Portal ↔ Grading Analysis	Navigation Portal allows Professor to edit/create the grading scheme of the Grading Analysis for lab/course	sends, recieves, stores, updates
Grading Analysis ↔ Simulation	Grading Scheme determines the point allocation for Simulation by updating the number of mistakes recorded	updates, records
Grading Analysis ↔ User List	Grading Scheme determines how the lab will be graded and therefore, update the students' corresponding grades in the User List as they move through the lab	updates
Controller ↔ Database Connection	Controller records the progress and each step that the user makes in the Simulation, and updates their corresponding grades	sends, updates, records
User Management ↔ User List	User Management lets the Professor update the User List	updates
Controller ↔ User Account	Controller updates the User Account	updates
Controller ↔ User Verifier	Controller requests user verification from User Verifier	request, verifies?

Figure 15: Association Description

When a student submits an answer during the simulation the simulation will communicate with the controller to verify the correctness of the students answer. The controller will record a students progress and relay the information to database to store the information. The controller will also communicate with the user verifier to facilitate user verification. User account will communicate with the controller to update the user accounts. The user verifier and the user list will communicate with each other make sure all the users are registered correctly. Users navigate to other pages using the navigation portal. When a user selects a destination, the navigation portal will communicate with the controller to send the user to their destination. The navigation portal will communicate with the simulation to allow students to access the simulation when they request it. The navigation portal will also communicate with the grading analysis to allow the professor edit the grading scheme for the lab. The grading analysis will communicate with the simulation to determine the point allocation for the lab. Grading analysis will need to communicate with the user list in order to update the students grades. User management will communicate with user list to give the professor the ability to update the user list.

5.1.3 Attribute Definitions

R#	Responsibility	Attribute	Concept
R17	Know if the current lab section is completed	isSectionCompleted	Controller
R18	Know if the simulation is waiting for input from the user	isAwaitingAction	
R19	Know if network connection is broken	isNetworkDown	
R20	Know if the user is in the Navigation Portal	isInNavigationPortal	
R21	Verification of nonexisting user	isNewUser	
R22	Know if the answer or action done is correct	isCorrect	
R23	Record the progress and each step that the students make	trackProgress	User Account
R24	Student's or Professor's NetID or name	ID/Name	
R25	Student's Grade	grade	
R26	Enter Simulation Portal	gotoSimulation	Navigation Portal
R27	Browse through the grading options that the professor can choose	gradingOptions	
R28	Enter the Grades Portal	gotoGrades	
R29	Start the simulation	startSimulation	Simulation
R30	Start the animation	gotoAnimation	
R31	Give tips and confirmation of answers or actions after each section is completed.	giveFeedback	
R32	The number of mistakes allowed before a deduction is made	strikeLimit	Grading Analysis
R33	The point value deducted per strike.	pointDeduction	
R34	The maximum number of strikes before the student is forced to repeat the simulation.	maxStrikeLimit	
R35	The network address of the relational database	dBnetworkAddress	Database Connection
R36	Questions and/or lab directions to follow	labQuestion	Lab Information
R37	The view of the microscope	adjustView	
R38	Create and edit a graph	createGraph	
R39	Create and edit a data table	createTable	

Figure 16: Attribute Description

The isSectionCompleted attribute verifies if the section of the lab simulation is completed. The isAwaitingAction attribute lets the system know it is waiting for information from the user. In order to verify that the network connection is available, isNetworkDown checks the connection. IsinNavigation portal checks if the system is in the navigation portal. When a new user is registering, isNewUser authenticates that the user is not already in the user database. When a student enters an answer, isCorrect checks if the inputted answer by the student is correct.

The system allows for the student to begin the lab where they left off because of the trackProgress attribute. The user name for the student or professor is required for the database to verify a registered user, the attribute for the name of the user is ID/name. The grade the student receives in the lab is stored in the grade database, the attribute for the grade is grade. The attribute which allows the student to enter the lab simulation is goToSimulation. The professor can choose criteria on how wrong answers will affect students grades on the simulation in the attribute gradingOptions. The attribute goToGrades allows the user to access their individual grades, while allowing the professor to see the grades of all the registered students. The attribute which prompts the system to start the simulation is startSimulation. An opening animation will perform for the student illustrating important concepts of the upcoming lab in the form of the attribute startAnimation.

The attribute feedback will give conceptual to the student at important junctures of the simulation. This is the amount of strikes a student can receive before having points deducted from their grade for the simulation is adjustable by the professor, and it is in the form of the attribute strikeLimit. The attribute pointDeduction withholds points from the students final grade every time the student does

a wrong action on the lab simulation or gets a question wrong. In order to insure that the student is properly learning the required information from the simulation, there is a limit of how many questions they can get wrong before having to restart the whole simulation. This limit can be set by the professor in the form of the attribute maxStrikeLimit.

When information regarding grades or users is needed from the database this address is needed to connect the system to the database properly. The attribute dbNetworkAccess will connect the database to the system. Throughout the simulation the student will be asked questions to verify that they have learned the required information which will be in the form of the attribute labquestion. The attribute adjustView will allow the student to zoom in or out on the microscope to properly view the mitosis process in a sample that cannot be seen by the naked eye. The attributes createGraph and createTable will allow students to create or edit graphs and tables illustrating the lab simulation.

5.1.4 Traceability Matrix

Use Case	PW	Domain Concepts									
		Navigation Portal	Controller	Database Connection	User Verifier	User Management	User List	Grade Analysis	Simulation	Lab Information	User Account
UC1	8	x	x	x			x	x	x		
UC2	2	x	x	x	x	x	x				
UC3	2		x	x	x	x	x				
UC4	3	x	x	x			x	x	x		x
UC5	3	x	x	x				x	x		
UC6	2	x	x	x		x	x				
UC7	8		x	x				x	x	x	x
UC8	18	x	x	x				x	x	x	x
UC9	7		x	x					x		x
Max PW:		18	18	18	2	2	8	18	18	18	18
Total PW:		36	53	53	4	6	17	29	47	26	36

Figure 17: Traceability Matrix(concepts vs use case)

5.2 System Operation Contracts

System operation contracts for the Fully-Dressed Use Cases are seen below, as they pertain to the format here:

- Name: appropriate name
- Responsibilities: perform a function
- Post Conditions
- Pre-Conditions

UC8: Simulation

- Name: simulation
- Responsibilities:
 - Start the simulation as well as the introduction animation
 - Take input from user interacting with simulation
 - Send useful feedback
 - Store history in the form of the table of contents(see UI specifications)
- Preconditions:
 - Logged in and registered user
 - Went through the Navigation Portal to access simulation
- Postconditions:
 - Introduction animation was started
 - Student was able to perform lab experiment

UC7: Give Feedback

- Name: giveFeedback
- Responsibilities:
 - Analyze student action
 - Send message based off student action
 - If error, increment error count
 - If error count reaches threshold, take appropriate action
 - Return to previous screen
- Preconditions:
 - Logged in and registered user

- Went through the Navigation Portal to access simulation
- Student reached point in simulation to prompt giveFeedback
- Postconditions:
 - Helpful feedback was given to the student.

UC1: Navigation Portal

- Name: navPortal
- Responsibilities:
 - Allow the user to login
 - Show a menu/start page showing the options available to that user
 - Allow the user to access those options
- Preconditions:
 - Registered user
- Postconditions:
 - Student was able to access their previous grades
 - Professor was able to access User List
 - Student was able to access the simulation
 - Professor was able to get to the Grading Analysis

The operation contract shows the capabilities of each of system in each of the different use cases, as well as what is expected of the system. The pre and post-conditions explain the state of the system before and after the use case scenario, and the responsibility shows the functions necessary to reach those conditions.

6 Plan of Work

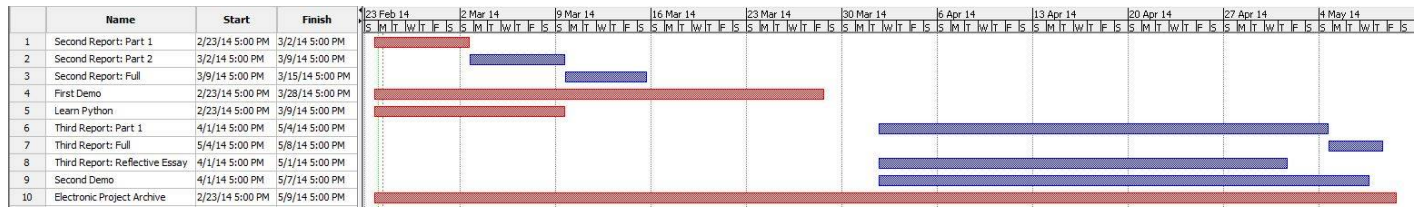


Figure 18: Timeline of the Plan of Work

With the submission of Report #1, the group will move onto writing Report #2 and developing the first demo. The demo and Report #2 have to be worked on simultaneously because the due dates for Report #2 are every 6-7 days following the Report #1 due date. Writing good quality code takes a while, so we will have to be on top of our time management. The real leg work begins when we start to write the software. As described in the product ownership, each pair of group members will work on the specified part of the code. However, since we are mostly inexperienced with python, we will begin by heavily focusing on learning the language. Our designated methods will be www.Codecademy.com, Learn Python the Hard Way, and other reading material. Because we all have some experience programming, at the very least C++, the process of learning python should not take more than 2 weeks. After we passed the tutorials, we will begin finding useful mods (ones that let us represent our code graphically).

After the first demo, which we will present on April 1st, we will start to work on Report #3-revising and improving from our past two reports. Along with the revisions for our report, we will also be revising our first demo to present as a second demo. This time, there are 3 parts that will be worked on at the same time. The first is the Part 1 of Report #3 is the main writing that the group has to work on, thus, it should be started right away. The second is the Reflective Essay which includes an individual write up and should be done on each group members own convenience. The third and final portion is the second demo. Since there is ample time between the first demo and the next due date, working on all three things simultaneously shouldnt pose too much of a problem, but we will not push off the time as there is still a lot to be done. The full report for Report #3 will be written immediately after Part 1. Because it is the last few days of class, we need the time to finish coding. So the report being finished sooner than later is crucial. Finally, the Electronic Project Archive can begin after the second demo. It is then that the coding should be accomplished. It should be finished last, since it compiles everything together.

6.1 Project Management

The entire project will be done through division of labor. The team will be divided into three groups of two. The lab experiment is broken up into 3 separate parts: The introduction learning simulation and navigation portal, the interactive simulation of mitosis, and the data collection tutorials. Each group will handle the User Interface, the interactive options, finding the graphics, creating the simulation, and providing a feedback system.

6.2 Introduction Simulation and Navigation Portal

The introduction simulation and the navigation portal will be created by Jeff Gillen and Evan Chipps. Their responsibilities include, but are not limited to, creating the simulation that is knowledgeable and easy to understand, allowing the students to go backwards if something does not make sense, and making it visually appealing. The navigation portal should allow the student to log-in, and give simple ways to navigate to the lab experiment and grades. The navigation should also allow the professor to access the lab, change the different settings of the simulation(such as grading system), and entire class grades.

6.3 Interactive Mitosis Experiment

Marvin Liu and Anthony Porturas will be responsible for the creation of the interactive simulation itself. This includes the realistic simulation of the process of cutting an onion root and placing it into a microscope to observe the plant cell, and to repeat the process with the animal cell. The system should have many options and include paths leading to common student mistakes. Depending on the action made, the system should respond and give real-time feedback to the student, so that the student can learn from the mistake. The system will keep track of the student's errors, and respond according the professor's policy.

6.4 Data Collection

The last objective listed in the lab manual states that at the end of the experiments, students should have a working knowledge of how to collect data through the use of graphs and tables. The system will include a graphing tutorial, created by Steve Pak and Brandon Lum. This data collection subsystem should introduce the concepts of graphs and tables, specifically the pie chart and bar graphs. The system should have a quick interactive tutorial, asking them to create both a pie chart and a bar graph based on a set of data. Incorrect graphs should give feedback to the student, saying that the student was wrong, and ask them to try again. Then the system should allow the students to create their own graphs and tables, based on their data from the experiment.

The above requirements for each pair ensures that each member of the group is playing a clearly defined and important role. They also ensure that each member is contributing fairly equally and no one member is taking on a dominant role. To make sure that the subsystems are not vastly different, each pair is responsible for updating the rest of the team. Although some collaboration is necessary for this project to be successful, no subsystem is dependent on the progress of another subsystem.

7 References

7.1 Biology-Related

Lab Manual <http://www.ece.rutgers.edu/~marsic/books/SE/projects/ViBE/biolab-1.pdf>

Biology Online <http://www.biology-online.org/>

7.2 Software-Related

FURPS+ <http://en.wikipedia.org/wiki/FURPS>

Virtual Bio Lab-2012 www.ece.rutgers.edu/~marsic/books/SE/projects/ViBE/2012-g5-report3.pdf

7.3 Pictures

Plants VS Animal <http://getyournotes.blogspot.com/2011/05/plant-and-animal-cells-differences-and.html>

Mitosis <http://royaleb.wordpress.com/2009/04/27/mitosis/>

Microscope <http://www.microscope-microscope.org/basic/buyers-guide.htm>

Hot Plate http://www.labdepotinc.com/Product_Details~id~604~pid~59646.aspx

Other Graphics https://github.com/sjlu/Virtual-Biology-Lab/tree/master/1_code/Flash/Virtual%20Bio%20Lab%20Graphics

Knife <http://www.teachersupplysource.com/images/D/034-AEP-7-372.jpg>

Whitefish <http://www.biologycorner.com/projects/mitosis.html>