



# Mobile DNA Sequencing Analysis

A Major Qualifying Project submitted to the faculty of Worcester Polytechnic Institute in partial fulfillment of the requirements for the Degree of Bachelor of Science.

**Advisor:**  
Patrick Flaherty

**Submitted By:**  
James Erickson  
Thomas Gammel  
Sam Miraglia  
Brie Newton

1 May 2014

## Table of Contents

Acknowledgments.....	3
Abstract.....	4
Table of Figures.....	5
Table of Tables.....	6
Chapter 1 Introduction.....	7
Chapter 2 Literature Review.....	8
2.1 DNA Sequencing.....	8
2.2 Next Generation Sequencing.....	11
2.3 Obstacles.....	13
Chapter 3 Project Strategy.....	15
3.1 Initial Client Statement.....	15
3.2 Objectives.....	15
3.3 Constraints.....	16
3.4 Revised Client Statement.....	16
3.5 Project Approach.....	17
3.5.1 Technical Approach.....	17
3.5.2 Management Approach.....	17
3.5.3 Financial Approach.....	18
Chapter 4 Alternative Designs.....	19
Chapter 5 Design Verification.....	21
Chapter 6 Discussion.....	27
Chapter 7 Final Design and Validation.....	29
7.1 Modeled Sequencer.....	29
7.2 iOS Application.....	30
Chapter 8 Conclusions and Recommendations.....	32
References.....	33
Appendix A – Gantt Chart.....	35
Appendix B- Work Breakdown Structure.....	36

## Acknowledgments

Our group would like to thank Professor Flaherty for all his assistance and enthusiasm throughout this project. Without his guidance this project would never have been possible. We would also like to thank Daniel Tocco for allowing us to use his Macbook as the primary device for this project and for his expertise as a programming consultant.

## **Abstract**

As technology used in everyday life becomes more mobile, the same can be said for DNA sequencing technology. There are many researchers and scientists searching for ways to make DNA sequencing technology more readily available for public use. However, as sequencing technology becomes more mobile, the analysis tools remain large and cumbersome. The design team was successfully able to create an application for iOS devices that receives data from a mobile sequencer and detects point mutations. The design team processed next generation sequencing data in .fasta format to identify mutations in the BRCA1 gene and display to a custom graphical user interface.

## Table of Figures

Figure 1. Cost of Next Generation Sequencing Over Time .....	10
Figure 2. First Stage in Next Generation Sequencing.....	11
Figure 3. Next Generation Chemistry Cycle.....	12
Figure 4. Typical NGS Workflow.....	12
Figure 5. Mobile DNA Sequencing Analysis Objective Tree.....	16
Figure 6. Proposed Software Pipeline.....	17
Figure 7. Screenshot of iPhone Chat Server Launching MATLAB .....	21
Figure 8. Screenshot of MATLAB Filtered Reads .....	22
Figure 9. Screenshot of Server Connection Request.....	22
Figure 10. Screenshot of Connected Device with Filtered Reads.....	23
Figure 11 Screenshot of Detected Mutations .....	23
Figure 12. K value vs. Time consumption per read of MATLAB filter .....	25
Figure 13. K value vs. Percent Throughput of Reads .....	25
Figure 14. Total time of Processing vs. File Size of MATLAB filter .....	26

## Table of Tables

Table 1. Pairwise Comparison Chart .....	15
Table 2. Financial Breakdown .....	18
Table 3. Design Alternatives.....	20
Table 4. Average Time and Memory Consumption from Varying File Sizes .....	24

## Chapter 1 Introduction

The idea of personal genome sequencing for diagnostic analysis has been far out of reach for the majority of the population since its inception. Through years of research, the Human Genome Project (HGP) successfully determined the order of nucleotide bases in the human genome. Since the HGP, organizations such as ENCODE Project and 1000 Genome Project have endeavored to discover the role of each gene and how differences and mutations affect a person's phenotypes. The use of genome sequencing for the early diagnosis of diseases has the potential to completely revolutionize how patients are screened for certain disorders such as cancer.

The cost of sequencing the human genome has dramatically decreased in recent years, however the analysis of the genome remains very expensive. It is irrefutable that the decreasing cost of sequencing itself will bring the benefits of genomic testing to entirely new demographics. However, without cost effective analysis tools the benefits of this technology will still only be available to a select group of people.

In addition to becoming more affordable, genome sequencing technology is also trending towards becoming more mobile. Personal sequencers are primed to hit the market in the coming years, allowing the general population to sequence genomes quickly and conveniently. In an ideal situation the user could then analyze their genome from the comfort of their home and then consult with their physician. The reality is that this data needs the processing capabilities of a supercomputer to perform this crucial analysis step. The full potential of this technology cannot be harnessed without a readily available sequence analysis platform for personal use.

The design team set out to create a mobile analysis platform to complement the convenience of mobile sequencing technology. Many people around the world possess some type of mobile phone and the proposed design would run on the iOS platform to provide the world with the ability to analyze the data output from a mobile sequencer. The design will begin with modeling a mobile DNA sequencing device that could send its data directly to an application dedicated to the analysis of a specific gene. The application will then be designed with the ability to stream the output data from the sequencing device and identify any mutations that may correspond to a hereditary mutation or genetic disease.

Throughout the following chapters we will discuss the background information pertinent to the project, including but not limited to how sequencing has become cheaper while analysis remains very expensive. The team will address the obstacles faced throughout the design process, and the approach taken to accomplish our goals. The team will then elaborate on the final design, its verification and validation, the results and final conclusions of the project. Finally, the group will propose recommendations for future work.

## Chapter 2 Literature Review

### 2.1 DNA Sequencing

DNA sequencing is the process of determining the order of nucleotide bases in a person's unique genetic makeup. The determination of these patterns has advanced applications in biological research and discovery. This technology has allowed scientists and researchers to translate the complex code of DNA into interpretable patterns and document the findings in what is known as the 'Human Genome Project'. The sequence of an individual's DNA gives specific phenotypic information including expression of common harmful mutations. The HGP has determined that there are about 20,500 human genes that code for specific biological characteristics in each individual [1].

Two examples of specific regions of the human genome are BRCA1 and BRCA2. BRCA1 and BRCA2 are naturally occurring human genes that produce tumor-suppressing proteins that help correct damaged DNA information [2]. Both of the genes play an important role in the regulation of cell proliferation. BRCA1 and BRCA2 mutations account for about 20-25% of hereditary breast cancers and about 5-10% of all breast cancers [3]. Additionally, mutations in these two genes can account for 15% of cases of ovarian cancer [3]. The National Cancer Institute reports that 55 to 65 percent of women who inherit a harmful BRCA1 mutation and 45% of women who inherit a harmful BRCA2 mutation will develop breast cancer by the age of 70 [3]. If this mutation is found early, preventative surgery can be performed that will increase patient survival rates significantly. This project aims to help expand the amount of women that could benefit from additional preventative diagnostics.

The two most common preventative surgeries are bilateral prophylactic mastectomy, or subcutaneous mastectomies [4]. Both surgeries have benefits, but the total mastectomy (total breast removal) provides the greatest breast cancer risk reduction. A report by cancer.gov states that bilateral prophylactic mastectomies are shown to reduce the risk of breast cancer by at least 95% with women who have deletion mutations in the BRCA1 or BRCA2 genes. Analysis of an individual's DNA sequence can be used as a diagnostic tool to identify harmful mutations such as these.

The genetic makeup of humans is very complex and the digital files that correspond to the DNA sequence data of a human genome require large amounts of computer storage. A .fastq or .fasta file is the standard data file output used for the majority of next generation DNA sequencing storage. Illumina, the current leader in next generation sequencing has machines that can sequence an entire genome in about 2 weeks. It outputs .fastq and .fasta files corresponding to the genomic data that can exceed 200 GB in size. The project design will consume a healthy breast cancer (BRCA) reference .fasta file as well as a sample of genomic data that needs to be processed. The .fasta files are incredibly complex in addition to their size, which is a primary design concern when considering application accuracy, processing requirements,



and speed. The limited resources that are available on a mobile device platform make string-based processing methods difficult to develop. The focus of the project will be to develop an iPhone application that requires minimal strain on the mobile processor or battery, while still accurately performing a DNA analysis on a .fasta file. This will provide a preliminary diagnostic of whether or not the sample has a mutation present.

DNA sequencing and analysis is currently only available to a very limited demographic due to the advanced technology and resources required. The development of a mobile platform would provide the benefit of early detection for genetic disorders to a wider population. Presently, the genome is only processed in its complete form and this analysis is not possible on a mobile platform. The application design is tailored around identifying specific regions of interest within the stream of an entire data set. This software pipeline will be optimized to reduce processing requirements and computational time on the mobile platform, while remaining easy to use and accurate enough to detect mutations in known genes.

Early detection of any condition can be particularly difficult when the disease shows no noticeable symptoms in its early stages. By using the data collected in the Human Genome Project (HGP), a medical professional is able to ascertain warning signs from an individual's DNA which correspond to a possible genetic condition. The HGP set the stage for genomic-based medicine when researchers around the world began to determine the DNA sequence of the entire human genome. Not only did this group set out to sequence the entire genome, but the end goal was to create a map that would show the connection between genes and their corresponding phenotypes. After years of data collection and analysis, the HGP researchers were able to successfully determine the order of all nucleotide bases in the human genome, map out specific locations of genes in chromosomes, and track inherited traits through family lines via the creation of linkage maps [5]. The research done on the HGP has and will continue to lead to improvements in the quality of medical care around the world.

Genetic information is a valuable data resource for medical professionals diagnosing commonly studied genetic diseases. As genetic factors are studied more thoroughly by both scientists and clinicians, these theories will be put to the test for many complex diseases affecting populations both large and small. Most treatment options based on genomic analysis are estimated to be 10 to 15 years away at the very least, but improvements in diagnostic testing should occur much sooner [6]. Figure 1 below shows the decreasing cost of sequencing an individual's DNA from 2001 to 2014. The green dotted line represents the cost of sequencing, while the white line represents Moore's Law. This relationship illustrates that the cost of sequencing is decreasing at such a rapid pace that the resulting data simply cannot be processed at a rate that is comparable.

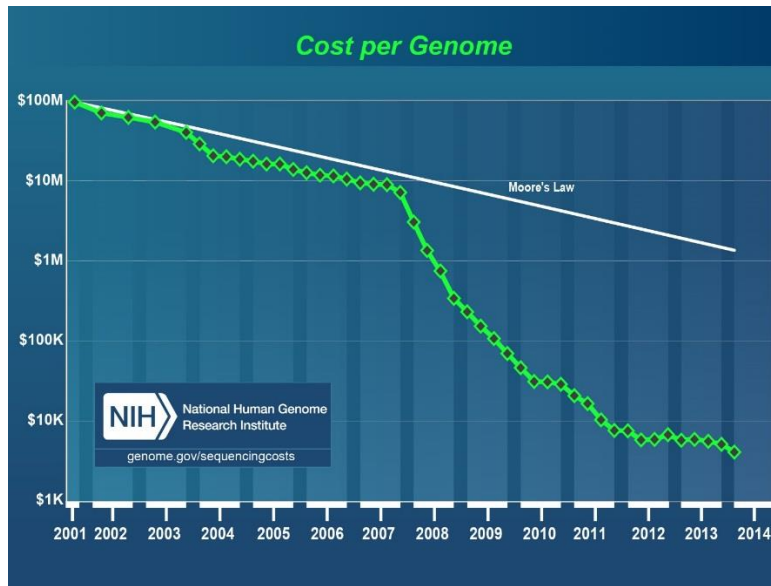


Figure 1. Cost of Next Generation Sequencing Over Time [7]

Early diagnosis is key to successful treatment with many forms of cancer. Genomic analysis allows a patient to be declared “at risk” years before a symptom is present. Genetic research looks to provide medical professionals and scientists with an understanding of hereditary risks associated with diseases, as well as how different elements work together to have profound effects on the entire human body [6].

The human genome is over 3 billion nucleotide bases in length and mapping of gene indices is necessary in order to effectively make use of this full sequence compiled by the HGP [8]. Genetic mapping was able to confirm that diseases transmitted from parents to their children are linked to genes and isolate single genes responsible for these disorders [9]. This mapping procedure was accomplished by comparing multiple DNA samples in order to identify common patterns within a sequence. The pattern was logged as a marker, or polymorphism, on the genome if these patterns were seen amongst family members sharing a specific disease or trait [9]. As the collection of these polymorphisms became more extensive, it became apparent that specific locations were directly related to diseases or traits.

## 2.2 Next Generation Sequencing

There have been many strides in improving sequencing technology since the mapping of the human genome. Currently sequencing done by Illumina is the most widely used platform and can be considered the industry standard [10]. The Illumina platform is considered Next Generation Sequencing (NGS) technology. The first step in this sequencing process can be seen in Figure 2 and involves creating dense clusters of fragmented DNA in channels of the flow cell [10].

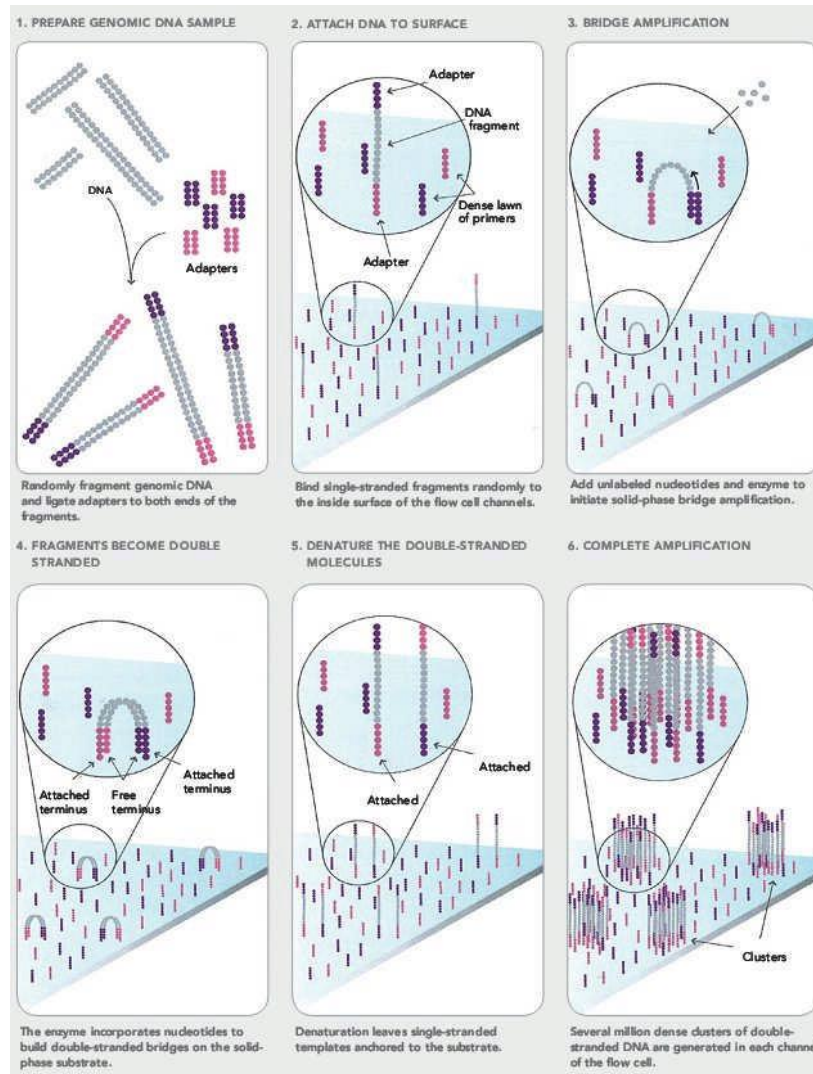


Figure 2. First Stage in Next Generation Sequencing

The next phase in NGS technology uses reverse terminators to allow the ability to read individual base pairs as they are added to the sequenced DNA, the process can be seen in Figure 3 below [10].

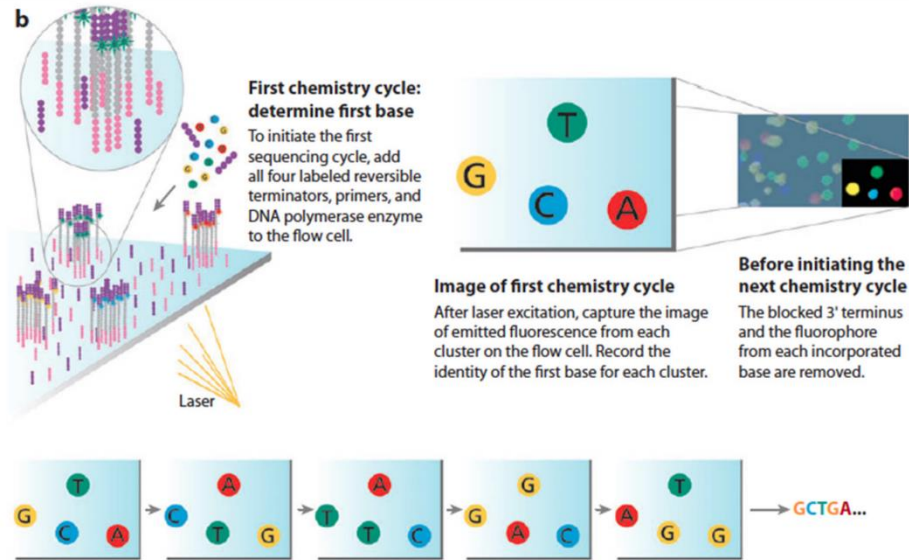


Figure 3. Next Generation Chemistry Cycle

The typical NGS workflow can be seen in Figure 4 which diagrams the entire process. One problem Illumina faces is a low multiplexing capability of large numbers of samples.

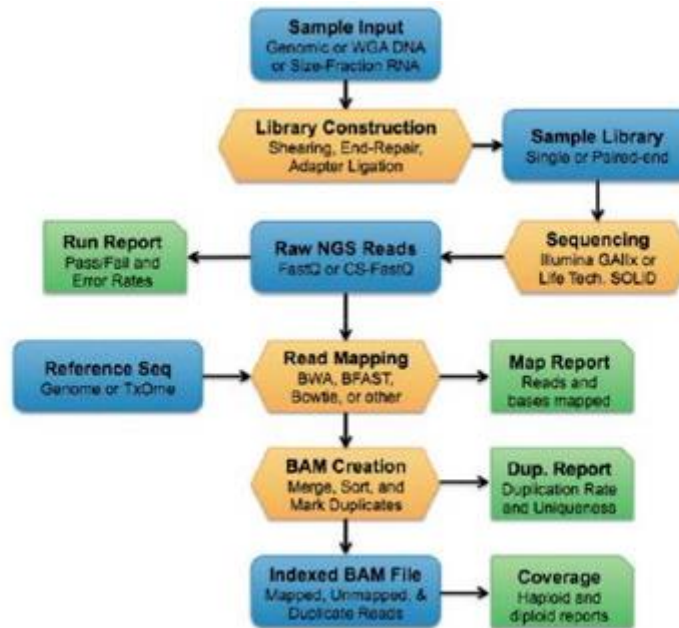


Figure 4. Typical NGS Workflow

Currently Illumina charges \$43 to sequence a gigabyte of data with intentions of bringing the price down to about \$29/GB in the near future [11]. The total price for the sequencing aspect is approximately \$44,000 for the combined generation, storage, and computation of the approximate 200GB of raw data. Researchers are successfully developing software tools to automate the sequencing process in

order to combat the current high cost of DNA sequencing [12]. The emergence of better sequencing instruments, robotics, and automation will further decrease the cost of genome sequencing [13].

Little work is being done to reduce the price associated with the actual analysis of the genome despite the downward trend in cost of sequencing. Without this analysis there is essentially no benefit to sequencing the genome in the first place. Oxford Nanopore technologies is currently developing a new generation of electronic systems that will detect genetic information on a mobile platform known as the MinION. The MinION is a disposable device that uses nanopores to perform real time genetic sequencing and can be plugged directly into a computer using a USB port [14]. This device could theoretically be partnered with the proposed novel design within the project in order to complete the analysis section that the MinIon cannot perform.

### 2.3 Obstacles

Many obstacles stand in the way of genome-based diagnostic medicine becoming readily available within the next few years. A majority of these stem from the fact that sequencing the human genome results in an extremely large amount of output data. DNA sequencing devices from Illumina and other NGS initiatives can output anywhere from 30 gigabytes to 1 TB of data for a single human genome [15]. An individual's DNA sequence is unique and confidential, and therefore must be properly stored and secured to prevent unauthorized access. Storing this amount of data presents a challenge with regards to storage cost per gigabyte, in addition to data security. Another major obstacle is in the large processing power required for computational analysis of the data. Currently supercomputers are the only way to process this data in a reasonable amount of time, but as technology becomes more sophisticated other avenues may become feasible.

Many of these challenges can be overcome with the proposed capabilities of emerging technology such as cloud computing. The International 1000 Genome Project has a catalog of more than 50 TB worth of DNA data to date [16]. If someone were to download this catalog they would not only need the extra 50+ TB of storage but it would take nearly 5 days to download it at an average download speed of 1 gigabit/second [17]. The concept of cloud computing suggests that it may be able to provide a solution by using the combined power of a network of computers over the internet. This network could then have the required computing power to tackle such an enormous dataset for the purposes of both storage and analysis.

With cloud computing for scientific applications still in its infancy, it is possible that the "big data" problem could be solved by simply focusing on smaller portions of this data, such as individual genes. An approach such as this could set the stage for mobile computing platforms to complement mobile sequencing technologies akin to the MinIon. While novel, this approach is not without unique

challenges of its own. There is no commonly used method for locating a particular gene within a completely unanalyzed genome and even then this gene's size may still pose a problem for the limited computing power of mobile devices. The proposed design offers the option for constant filtering of an incoming genome .fasta sample, and this pre-filtering stage can be perfected on devices such as a MinIon in order to make this process feasible.

## Chapter 3 Project Strategy

### 3.1 Initial Client Statement

To design and implement a software application that runs on a mobile platform and can accurately call specific alleles relevant to a genetic mutation from next generation sequencing data.

### 3.2 Objectives

The entire design process for this DNA sequencing application is tailored around five prioritized objectives. The objectives are: portability, accuracy, efficiency, speed, and user-friendliness. After multiple discussions between the project group and the client, these objectives were ranked using a pairwise comparison chart. This chart is shown in Table 1. The final design for the application should be *efficient* in optimizing CPU usage and battery life, with *quick* computational time and *accuracy* in mutation searching, all while remaining *portable*. The end goal of the project is to help provide the capabilities of next generation sequencing analysis to settings such as developing countries or classrooms studying genomics.

	Portable	Accurate	Efficient	Quick	User Friendly	Rank
Portable		1	0	1	1	3
Accurate	0		0	0	1	1
Efficient	1	1		1	1	4
Quick	0	1	0		1	2
User Friendly	0	0	0	0		0

**Table 1. Pairwise Comparison Chart**

Figure 5 below depicts an organized graphic portrayal of the group's primary design objectives and sub-objectives after communicating with the client.

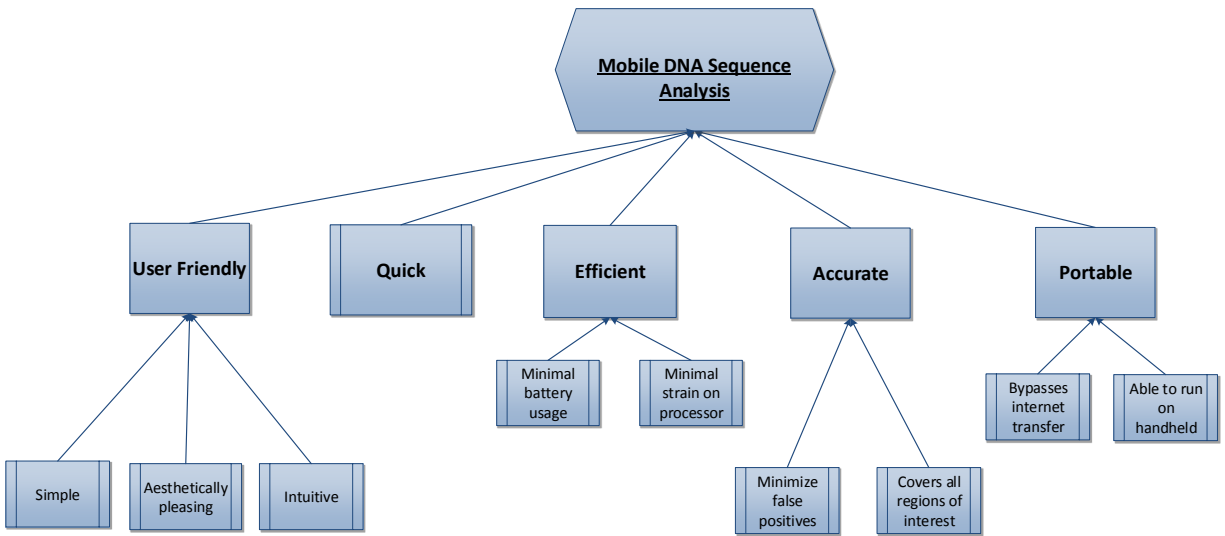


Figure 5. Mobile DNA Sequencing Analysis Objective Tree

### 3.3 Constraints

The major constraint associated with the design solution pertains to the output file size. The file coming off the sequencer to be processed by the application is anticipated to be about 200 GB in size and the largest iPhone has only 64GB of memory. This large file size causes a considerable data transfer obstacle due to the limited internet access in third world settings. There is no “standard” hospital room or equipment in a third world setting, meaning that the device must not rely on any specific secondary device or prior knowledge for the user. The maximum desired run time that the design team has placed as a constraint is roughly 1 hour, while completely and adequately analyzing DNA sequence data for abnormalities.

### 3.4 Revised Client Statement

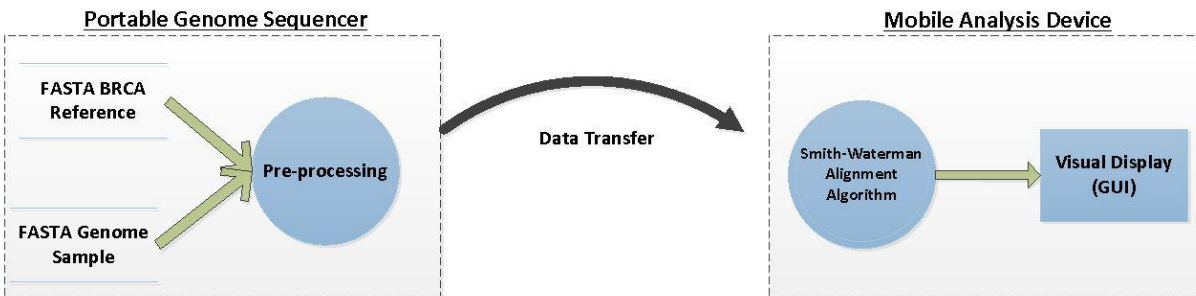
The design team conferred with the client and conducted independent literature searches to further understand the project and its challenges, subsequently revising the client statement to clearly define the needs.

The project group will develop a robust, accurate analysis software pipeline for stream-based processing of DNA sequence data. The software will run on an iOS platform and will process a targeted portion of the sequence data in FASTA-format.



## 3.5 Project Approach

### 3.5.1 Technical Approach



**Figure 6. Proposed Software Pipeline**

Figure 6 above shows the predicted strategy of how the group anticipates the DNA data will flow through our software design. The team used a divide and conquer approach in two parts: the simulated genome sequencer and the mobile analysis device. The simulated genome sequencer is modeled in order to test functionality of the mobile analysis and many assumptions are made about its functionality. The sequencer is not only capable of performing NGS techniques to produce output in .fasta format, but also performing filtering in order to compile reads relevant to a particular gene (BRCA1). Server capabilities are also assumed for the sequencer, in that it can connect to a mobile device and transfer filtered reads for alignment and analysis. The other end of the envisioned server connection is the mobile analysis application itself. The team will be designing this application to receive a stream of relevant, filtered reads from the sequencer and then align these along a reference gene to identify point mutations. All point mutations will be displayed in a final plot that illustrates the number of mutations at each index. Each index will display the type and frequency of each mutation on a stacked bar chart.

### 3.5.2 Management Approach

The project will take place over four, seven-week terms with the majority of the project taking place in the middle two terms. The project will consist of creating a platform, documentation, and presentations. The timeline of the project is diagrammed in the Gantt chart found in Appendix A. The group will be having weekly advisor meetings to assure communication and MQP objectives are met. The group will also complete progress reports every 4 weeks to document progress in addition to maintaining an electronic lab notebook including all developed code. It is one of the team's goals to assure that all four team members equally contribute to all parts of the project.

In addition, the group created a work breakdown structure to clearly divide the project into phases and a subdivision of effort to achieve each phase. The work breakdown structure can be found in Appendix B and contains both required steps in completing the project (solid lines) and additional desired steps (dashed lines) once the required goals are met.

### 3.5.3 Financial Approach

This project is mainly a software development project meaning the budget is minimal. The team's estimated expenses can be seen below in Table 2. The Apple developer's license will allow the team to push the application to the Apple app store if desired. The MacMini will provide the group with the physical machine needed to design and implement software in Xcode, as it is an Apple specific development environment.

<b>Possible Expenses</b>		
Apple developer's license		\$99
MacMini		\$200
<b>Total</b>		<b>\$299</b>

**Table 2. Financial Breakdown**

## Chapter 4 Alternative Designs

A number of criteria must be met in order to create a functioning system for the purpose of designing a mobile method of genomic sequence analysis. In short, the system must process a small section of the human genome and search for abnormalities as compared to a healthy reference gene. These abnormalities must then be presented to the user in such a way that possible mutations can be identified and isolated for further analysis and diagnosis.

The design team set out to create an iOS application that works in conjunction with a theoretical handheld gene sequencer to search for abnormalities in known genes specifically contributing to the cancerous proliferation of breast cells. Due to the extremely limited processing power of mobile devices in terms of memory and storage, the system was designed to conduct analysis focused on one single gene such as BRCA1 as opposed to the entire human genome. The theoretical sequencer would be designed to handle two major steps in the overall system. After the genome is sequenced, it must filter its own reads in .fasta format to produce a set of reads pertinent to the gene of interest. Once this filtering is completed a file is then transferred to the iPhone containing all pertinent reads from the filter. The iPhone receives a stream of this information and aligns these reads with a reference gene sequence, such as those available in genomic databases, in order to identify mutations. All possible mutations are recorded and plotted in a custom GUI, in order to provide a visual representation of mutations that occurred at each index of the reference gene.

Table 3 below depicts the design alternatives considered throughout the project. The client suggested that we use a suffix tree or a string filtration algorithm to accomplish the desired pre-processing. A suffix tree is a very advanced predictive search tool that uses a hierarchical structure to create an efficient selective filter. String filtration may be less efficient by comparison, however is straight forward for rapid implementation. Options for data transfer include both wireless and wired options such as USB, Bluetooth, and creating a socket server for the simulation. The alignment process can be accomplished using a number of different algorithms commonly utilized in the bioinformatics field. Finally, there are a number of plotting frameworks available for the iOS platform that are both free and available for purchase.

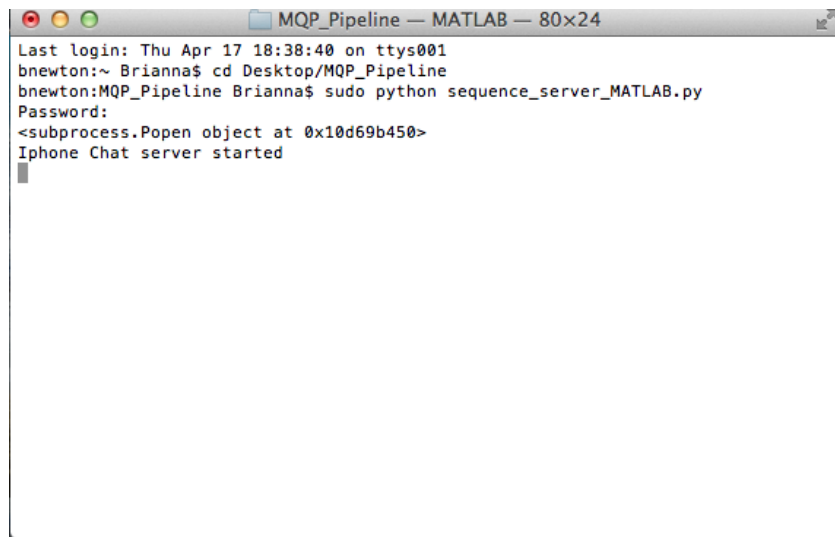
Table 3. Design Alternatives

<b><u>PRE-PROCESSING</u></b> Suffix Tree String Filtering	<b><u>DATA TRANSFER</u></b> USB Bluetooth Socket Server
<b><u>ALIGNMENT</u></b> BLAST Eland RMAP Smith-Waterman	<b><u>GUI</u></b> PowerPlot ShinobiControls Custom Core Plot

## Chapter 5 Design Verification

The final design demonstrates that large .fasta files can be accurately filtered and aligned for possible mutations using mobile technologies. NGS reads can be pre-filtered on a mobile sequencing device to produce a set of reads relevant to a particular gene before transfer to a mobile phone or tablet for final alignment. These mutations can then be displayed in an intuitive and visually appealing manner for the user.

Figure 7 below shows a screenshot of the server that calls the MATLAB filter. This is the first step that will run on the simulated sequencing device.



```
MQP_Pipeline — MATLAB — 80x24
Last login: Thu Apr 17 18:38:40 on ttys001
bnewton:~ Brianna$ cd Desktop/MQP_Pipeline
bnewton:MQP_Pipeline Brianna$ sudo python sequence_server_MATLAB.py
Password:
<subprocess.Popen object at 0x10d69b450>
Iphone Chat server started
```

**Figure 7. Screenshot of iPhone Chat Server Launching MATLAB**

Figure 8 below shows the MATLAB pre-processing stage that consumes the healthy reference BRCA file as well as a raw data sample. The command window prints the filtered reads that are considered relevant to the BRCA reference.

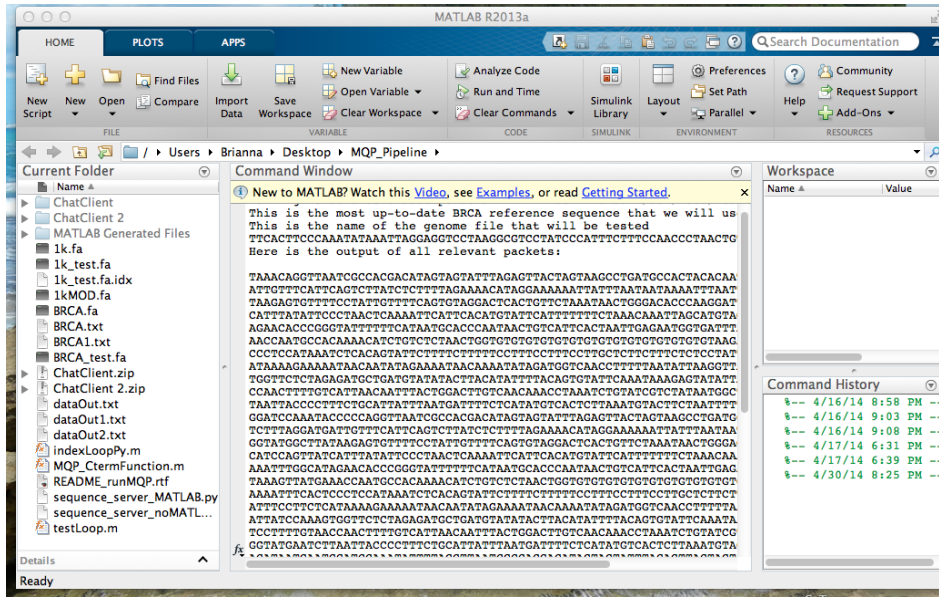


Figure 8. Screenshot of MATLAB Filtered Reads

Figure 9 below shows the iPad application attempting to establish communication with the server.

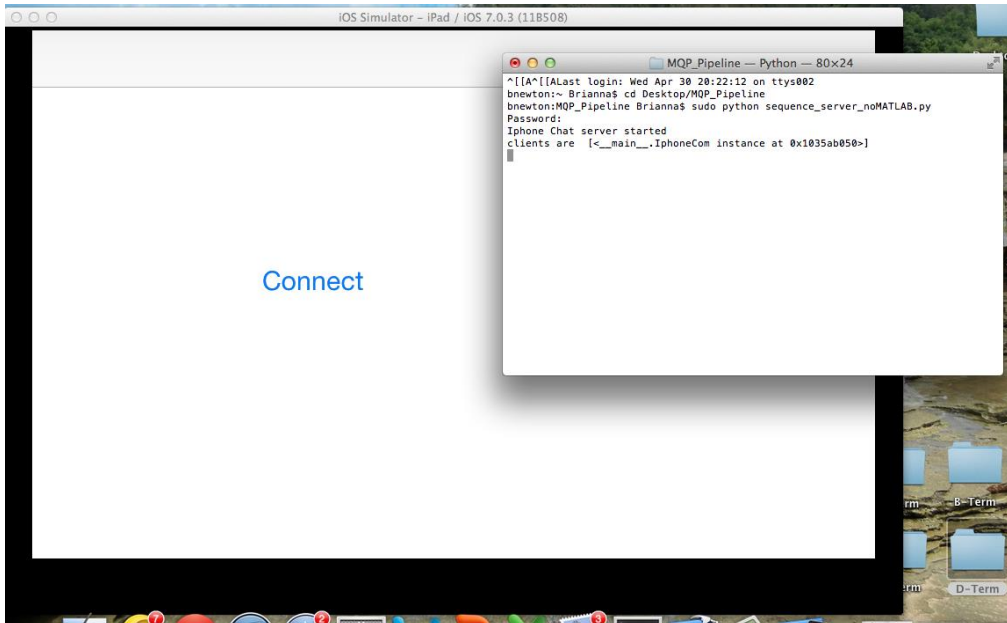


Figure 9. Screenshot of Server Connection Request

Figure 10 below shows a successful connection between the simulated sequencer and the iOS device. Each line of text corresponds to a relevant filtered read.

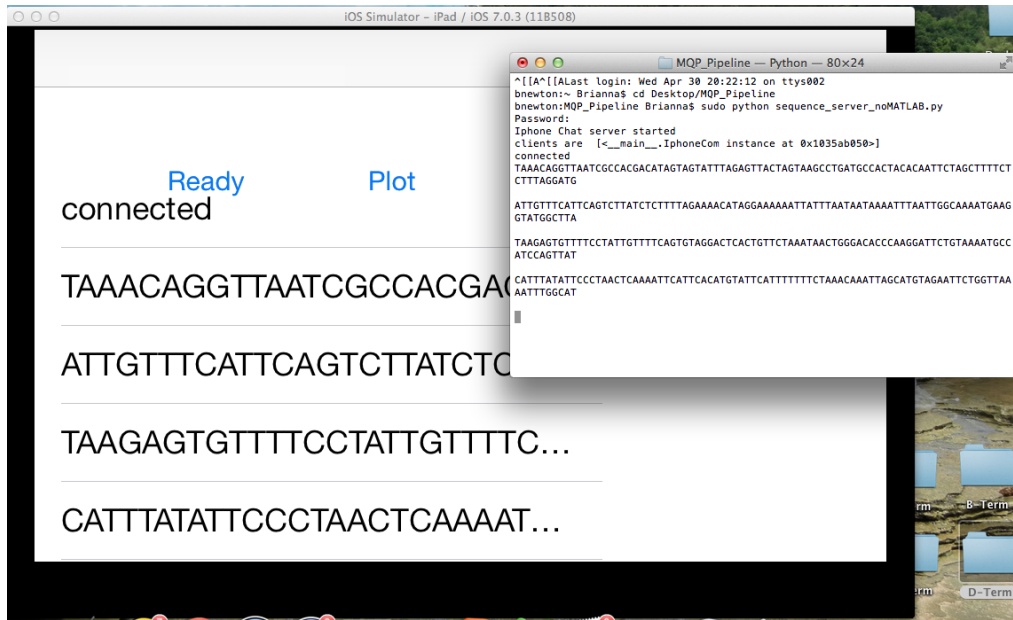


Figure 10. Screenshot of Connected Device with Filtered Reads

Figure 11 below displays the final screen of the application. The x-axis corresponds to the location of the indices along the reference BRCA gene. The y-axis corresponds to frequency of detected mutations. Each color represents a different nucleotide base mutation.

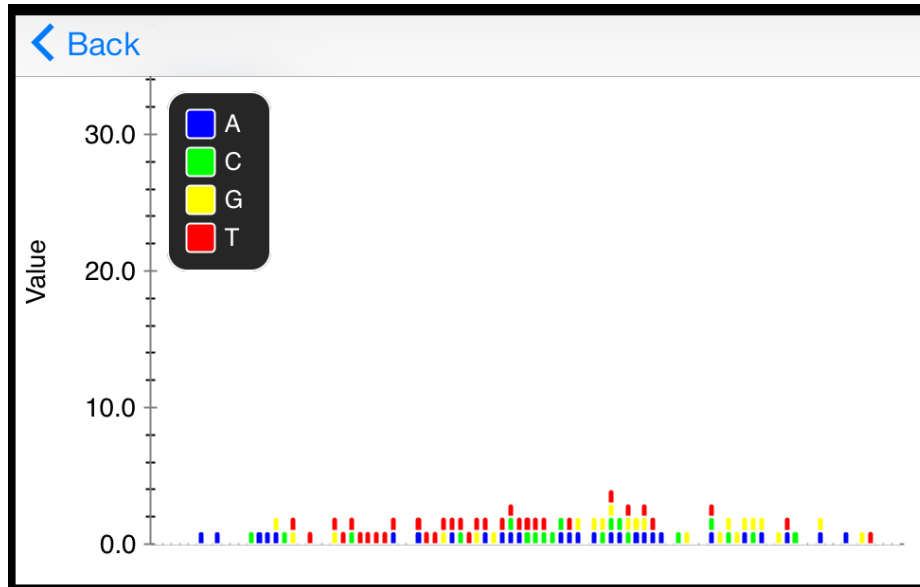


Figure 11 Screenshot of Detected Mutations

The design features below assess the quality of the pre-processor. Table 4 below outlines the average time and memory consumption of small, medium, and large sized files. The segment of code that loads and initializes the algorithm is compared to the entire pre-filtering design by using the other segments of code as a reference. Each time corresponds to the average time that it took to process one read, and is taken by using the tic, toc functionality within MATLAB. This function times how long the written code takes from tic to toc. All results were collected on the same computer on a WPI desktop. The beginning time for the code takes approximately 0.55 seconds to initiate, and the average time per read lower in the code within the for loop reveals that a read takes approximately 3.91 seconds. Considering that the genome file is only one read long for simplicity's sake, we can determine that the for loop filtering, processing, and saving accounts for the majority of the time of the algorithm. The second column in the table shows the average memory consumed for each read.

**Table 4. Average Time and Memory Consumption from Varying File Sizes**

	<b>Time</b>	<b>Memory (bytes)</b>
<b>load data</b>	0.551417	3,588,096
<b>small</b>	4.229084	2,260,173
<b>medium</b>	3.919017	542,474.2
<b>large</b>	3.589561	278,444.4

Figure 12 below shows the 'K' value vs. time per read of the MATLAB filter. The 'K' value corresponds to the number of consecutive reads that must match a healthy BRCA reference file in order to be considered relevant. The graph shows increasing 'K' values for the MATLAB filter and as 'K' increases the filter becomes more stringent. Figure 13 below shows this trend towards increased filtration of reads as 'K' increases. The y-axis shows the percentage of reads that were passed through the filter to the next stage of processing.



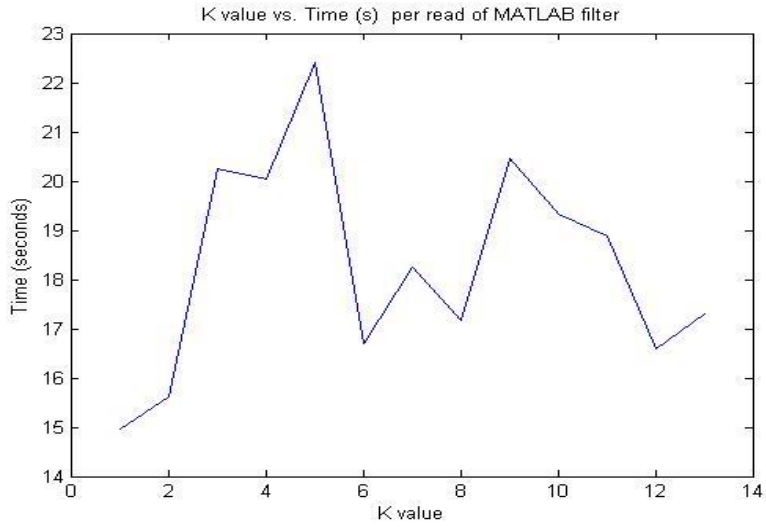


Figure 12. K value vs. Time consumption per read of MATLAB filter

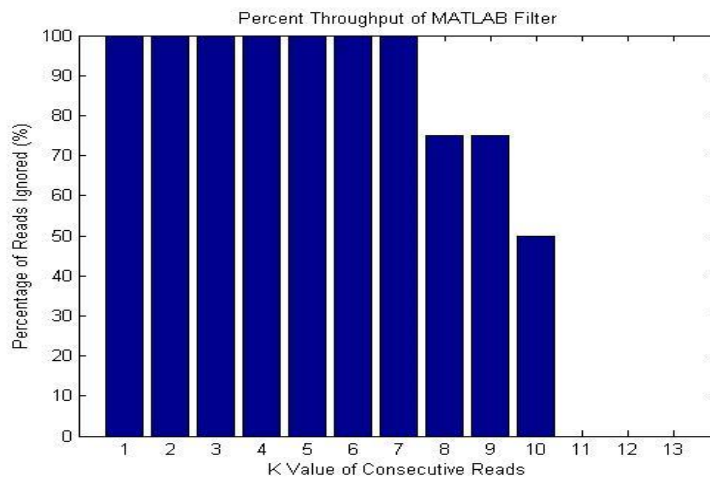
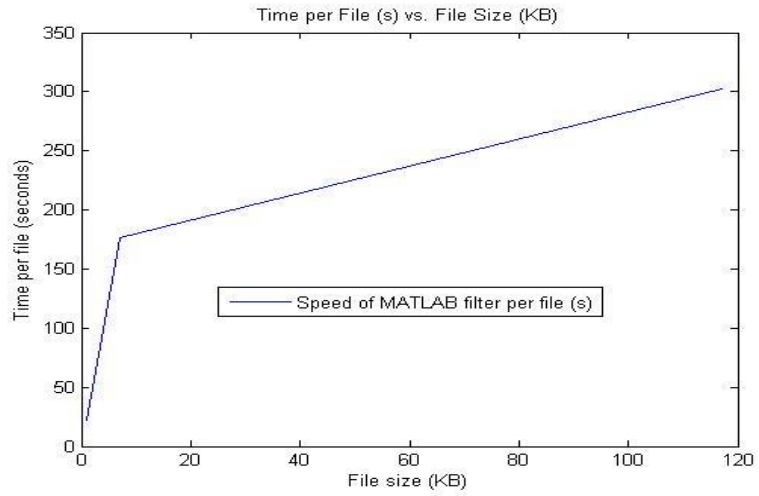


Figure 13. K value vs. Percent Throughput of Reads

Additional testing was performed on the total speed of the filtering as file size increases. Figure 14 below shows the total time of pre-processing for file sizes from 20KB-120KB.



**Figure 14. Total time of Processing vs. File Size of MATLAB filter**

## Chapter 6 Discussion

The design team successfully developed a connection between the pre-processing algorithm and the server. Figure 7 above confirms that communications were established, with Figure 8 illustrating a successful import of the BRCA reference and printed reads output by the filter. Since channels for communication are established upon startup of the server, the iOS application is able to retrieve pertinent reads by pressing the 'connect' and 'ready' buttons in sequence. The connection is confirmed in Figure 10, as identical reads appear on both the terminal window of the server and the user interface of the application. Once the reads have been received by the application, the user presses 'plot' to initiate the alignment process using the Smith-Waterman algorithm and all detected mutations are shown on the generated plot. This final plot, as displayed in Figure 11, provides the user and their physician with the locations at which mutations are present. Areas of the BRCA reference that show multiple stacked mutations are of greater concern than those displaying a single bar, as these are more likely to represent actual point mutations. Locations with very few mutations may correspond to an error inherent to the next generation sequencing process, not necessarily a harmful mutation. It should be noted that indices displaying all four nucleotides as possible mutations may indicate an insertion at this position, which should be treated in a manner different from that of a simple point mutation.

The main objectives of the design team when approaching this project were: portability, efficiency, accuracy, speed, and user-friendliness. When analyzing the entire software pipeline, the speed and accuracy are entirely dependent on the pre-processing conducted on the modeled sequencer. It is understood that the Smith-Waterman alignment algorithm will align any relevant reads with the provided reference sequence, but any reads accidentally filtered out with too strict of a filter will not reach this stage and will sacrifice the overall accuracy of the system. Conversely, if an excess amount of reads are allowed to pass through the pre-processing stage then speed and memory usage are sacrificed. This is an engineering tradeoff and in the interest of designing a mobile system, speed and memory usage take priority over accuracy during early development.

Table 4 above shows the average time and memory consumption of each read in a small, medium, and large file. The small file consists of five .fasta reads, the medium file consists of 25 .fasta reads, and the large file consists of 125 .fasta reads. All read times were measured using the tic toc functionality in MATLAB which measures the time from tic to toc. This data is compared to the beginning section of the code which loads the file and is outside of the filtering loop. The idea was to diagnose the slowest areas of code within the entire pre-processing filter. It is apparent that the average read time across all files is about 3.9 seconds per read which is nearly 90% of the total computational time assuming a sample file is one read long. Additionally, the memory column of the table proves that the maximum amount of memory that will be consumed is about 3,500,000 bytes. This area of code loads all .fasta files and

prepares them for string filtering. The information stored decreases over time as the filter removes non-relevant reads.

An entire sample is difficult to predict because it is entirely contingent on whether or not each read is considered 'relevant'. This relevance triggers the function to open a 'dataOut' text file, append to the end of the file, and re-save the file. This process is quickly skipped if the read is not considered relevant and passes in another read. The dependence on whether or not a read is relevant is the primary determinant on processing speed and memory requirements, and it is difficult to predict between different files. The memory consumption of each read is based on whether or not the read is relevant as well. Then furthermore it depends how large the 'dataOut' text file is existing on the same directory. The 'dataOut' file is reopened and appended to at the end of each relevant read, and therefore this file will continue to get larger and larger as the processing progresses.

Further testing was conducted to assess the dependence of the 'K' value on time and memory consumption. The project group tested the average time per read and the percent throughput of all reads against an increasing K value. Figure 12 above shows that increasing the specificity of the filter does not affect the average time per read in a predictable manner. Furthermore, Figure 13 above shows that 100% of reads are passed through the filter when K is less than or equal to 7. As expected, larger values of K filter all reads, while some discarded reads could possibly be related to the BRCA reference. After considering the design tradeoff between speed and accuracy, the group recommends a 'K' value of 8 or 9.

With regards to the "big data" problem, the size of files being sent to the mobile device was the primary concern. Encouragingly, Figure 14 illustrates a projected relationship between processing time and file size that appears to be logarithmic in nature. As file size increases, the processing time is expected to reach an asymptote. This maximum limit will allow for the recommendation of a system with an estimated minimum battery life. From the data collected, the relationship can be extrapolated to predict the time necessary to process a file in the range of 200GB.

As with any engineering design project, there will be profound effects on many aspects of society. With regards to economic impacts, this project stands to greatly reduce the number of surgical procedures performed due to an increased rate of early detection of genetic disease. Consequently, the cost of healthcare in general should decrease as well. By decreasing the overall cost of genetic testing and diagnosis, the benefits of this screening has the potential to affect people in all areas of the world and contribute to increased life expectancy. As with any technology that provides potential benefits to overall personal health, there are also issues that may arise as a result of successful diagnosis. The ethical conundrum that may occur entails a diagnosis being made, while the necessary treatment is not available. The question then is: does this diagnosis negatively affect the patient's quality of life?

## Chapter 7 Final Design and Validation

In order to develop an application to analyze a particular gene for point mutations, the design team had to create a model for a mobile sequencer and design the application's functionality to complement all of the sequencer's features. The team's final design consisted of a string filtration pre-processor, a socket server to transfer data, Smith-Waterman alignment algorithm, and custom Core Plot stacked bar chart to display mutations.

### 7.1 Modeled Sequencer

The first half of the software pipeline can be imagined as a mobile sequencer that will have the capability to identify reads from a specific gene of interest and stream this information to a mobile device such as an iPhone. The client had originally suggested the use of suffix trees algorithms to identify reads relevant to the gene of interest, but attempts to design this algorithm in Python were unsuccessful due to limited knowledge of the data structure and limited time. Following additional discussion with the client, it was decided that a similar string filtering algorithm could be written in MATLAB using an iterative process of matching reads to a reference gene for partial alignment.

The most basic functionality of the MATLAB algorithm is the string filter. This filter was built to read in a reference file with a length of characters that is about 10,000 characters long. This filter uses another input string of characters and compares this sample string of NGS reads to the reference file. Inputted FASTA files from NGS runs are stripped of their headers ('>chr1 50000'), leaving a simple text file containing unfiltered reads. This file can then be approached as a collection of strings in which a user-defined number of characters ('K') could be parsed and matched against the reference gene for partial alignment. Any and all reads that do not meet this partial alignment are effectively filtered out of the dataset that is sent to the mobile device. The filter looks for a successive match of user input 'K', and will output the entire read as a string output if this condition is met. The algorithm opens an output file called 'dataOut' and appends the relevant read to the end of the file. This simple filter functionality was expanded by writing an overarching function that can consume the name of a .fasta BRCA reference, as well as the name of a .fasta sample file. It then uses this information and the bioinformatics toolbox addition on MATLAB in order to read in the .fasta files one line at a time, and feeds them in to the base filter function. The proof of concept of this filter has proven successful on MATLAB software.

In addition to filtering, the sequencer should also contain server capabilities allowing for connection and communication with an iPhone, in which filtered data is transferred. After discussion with the client, a basic framework for this server was created in Python from a networking tutorial on creating an iOS chat application [17]. The server in the tutorial contains one main class for the iPhone communication protocol. This protocol includes class definitions for when a connection is established or lost, when packets of data are received by the server, and when a message is sent by the server. Twisted

framework is used to assign a port for communication and run the reactor outside of the communication protocol class. Twisted is an open source networking engine that simplifies the process of creating a server and its use was specified in the networking tutorial mentioned previously. While the server could have been created in a variety of other languages, the design implemented in the networking tutorial was basic and easily expanded to include the use of algorithms for filtering. For example, the original Python server code was modified to launch MATLAB's pre-processing algorithm described above. All of this functionality is envisioned to exist on mobile sequencing platforms of the future.

## 7.2 iOS Application

The second half of the software pipeline is an iOS application that can connect to and request data from the simulated sequencer through the server. The client required an application that could stream data from the server and this basic functionality was provided through the same networking tutorial referenced in the previous section. At this stage, the application was capable of connecting and communicating with the server through a graphical user interface (GUI) and displaying all messages on the same GUI on an iPhone. The communication with the sequencer is initialized using a 'Ready' button in the iOS framework in order to call each read in the filtered dataOut.txt file and transfer it to the iOS device.

Once the iOS device has received all reads from the server, it is then responsible for aligning this read with a reference gene to identify mutations. These reads are then passed through the Smith-Waterman algorithm when the user presses 'Plot' in order to align the segments and identify areas of possible point mutations. The client suggested the use of a Smith Waterman algorithm for the alignment stage of the software pipeline and through extensive research the team found an open source modified Smith Waterman algorithm that was proven to be ~50 times faster than its unedited form [18]. This algorithm was downloaded and modified further to meet the client's needs with regards to formatting the output for plotting. It was specified that the position of a mutation along the reference gene should be saved with both the mutated and expected nucleotide bases at that position.

The received reads will align to the most likely position on the BRCA reference gene, and undergo a comparison analysis that identifies point mutations with the use of the customized Smith-Waterman algorithm. The reads are transferred, aligned, and scanned through this algorithm, and four mutable arrays are created. A mutable array is created for each nucleotide base: adenine, guanine, cytosine, and thymine. Each one of these arrays contain the number of mutations at a given reference index corresponding with that base. A mutable array is more appropriate than an array of character arrays in this instance for multiple reasons. The character array would need to be initialized to a fixed size that could hold the largest number of mutations imaginable. Furthermore, an index cannot be represented as a character if the index of mutation is two digits or more.

The remaining step in the design of the iOS application pertains to the presentation of mutation data. These mutations should then be presented to the user in such a way that conclusions are easily reached. Rather than provide a table of this mutation data, the design team decided that it would be more intuitive to present mutations at specific indices in the form of a plot. A bar chart seemed to provide a structure most conducive to this type of data and many options exist within Xcode for creating plots such as this. The Core Plot library is an open-source plotting framework that provides 2D visualization of data and several plotting features. Core Plot is free to download and distributed with a Berkeley Software Distribution (BSD) license, as is Xcode. This framework is preferred over commercial frameworks such as ShinobiControls which costs \$995 per developer license [19], well outside of the project's budget. Core Plot is capable of displaying multiple collections of data on the same plot, with options for overlaying plots as well.

Once the mutable arrays are created, they are then plotted on top of one another in a stacked bar chart in order to display mutations at each index. The bar chart is created with the use of the Core Plot library for Xcode. Core Plot displays the mutable arrays by creating a plot space determined by the length of each array and maximum number of mutations at each index. The arrays are then graphed along the x-axis placing a bar at each index with the height signifying the number of mutations. The nucleotides are color coded where adenine is the color blue, cytosine is the color green, guanine is yellow, and thymine is red. Color bars will stack on top of each other if the algorithm aligns reads such that detected mutations overlap at specific indices on the BRCA reference. These data points are detected variances from the BRCA reference file that indicate point mutations from a healthy BRCA file. Once graphed, the user can easily identify areas with numerous mutations based on the height of each bar.

## Chapter 8 Conclusions and Recommendations

The proposed design successfully demonstrates that filtered files can indeed be realigned, and successfully display mutations on a stacked bar chart using a mobile platform for analysis. The software pipeline created by the design team incorporates a two part design serving as a proof of concept for mobile sequencing and analysis. This work in combination with strides towards developing mobile sequencers has the potential to inspire innovation in personalized healthcare. With this design, the team has created a basic platform for advancements in on-site genome sequencing and analysis in real-time. Successful demonstration of the idea suggests that improvements on pre-filtering will move towards a fully functional design that can be integrated with sequencing technologies. This novel approach strategically targets a small portion of the genome through the use of information provided by genetic databases and mapping. The group sees this technology being utilized throughout the world in a variety of settings. For example, the device could be used in a classroom to teach students about their own genetic makeup, as well as in a clinic in a developing country for on-site screening.

While this design demonstrates a proof of concept for mobile analysis, this is just the first step in establishing a marketable system. The MATLAB environment proves to be an adequate pre-filtering platform, but theoretical mobile analysis platforms will require a programming language better suited to embedded applications. Currently, there exists a python integration with MATLAB that calls the functions without launching the entire program and saves time and memory, but a full python integration is ideal for full compatibility with apple devices. This Python/MATLAB integration allows for the future design improvements to be tailored around moving away from the slower MATLAB environment altogether, and moving towards 100% python filtering. The filtering did not prove to be as fast as what was requested, and additional improvements will push the design towards optimal efficiency. Further testing incorporating different sample files will strengthen the group recommendation towards a specific 'K' value. Additionally, testing of file sizes above 120KB will improve the predictions for file sizes approaching 200GB.

If the device was to perform analysis over time on a regular basis, frameworks such as Core Data could be used to compact the mutable arrays and minimize memory usage and long term storage. However, due to the preprocessing of the data, this was not necessary at this stage in the project. Additional modules could also be expanded off of the Xcode application to connect to the cloud for more sophisticated processing or to email all results immediately to a physician. Consultation with a physician could lead to the recommendation of a threshold that predicts the probability of a genetic condition or disease. Lastly, the design can be easily modified and scaled to detect mutations in other genes of interest such as BRCA2 or TP53 by simply swapping out the reference sequence.



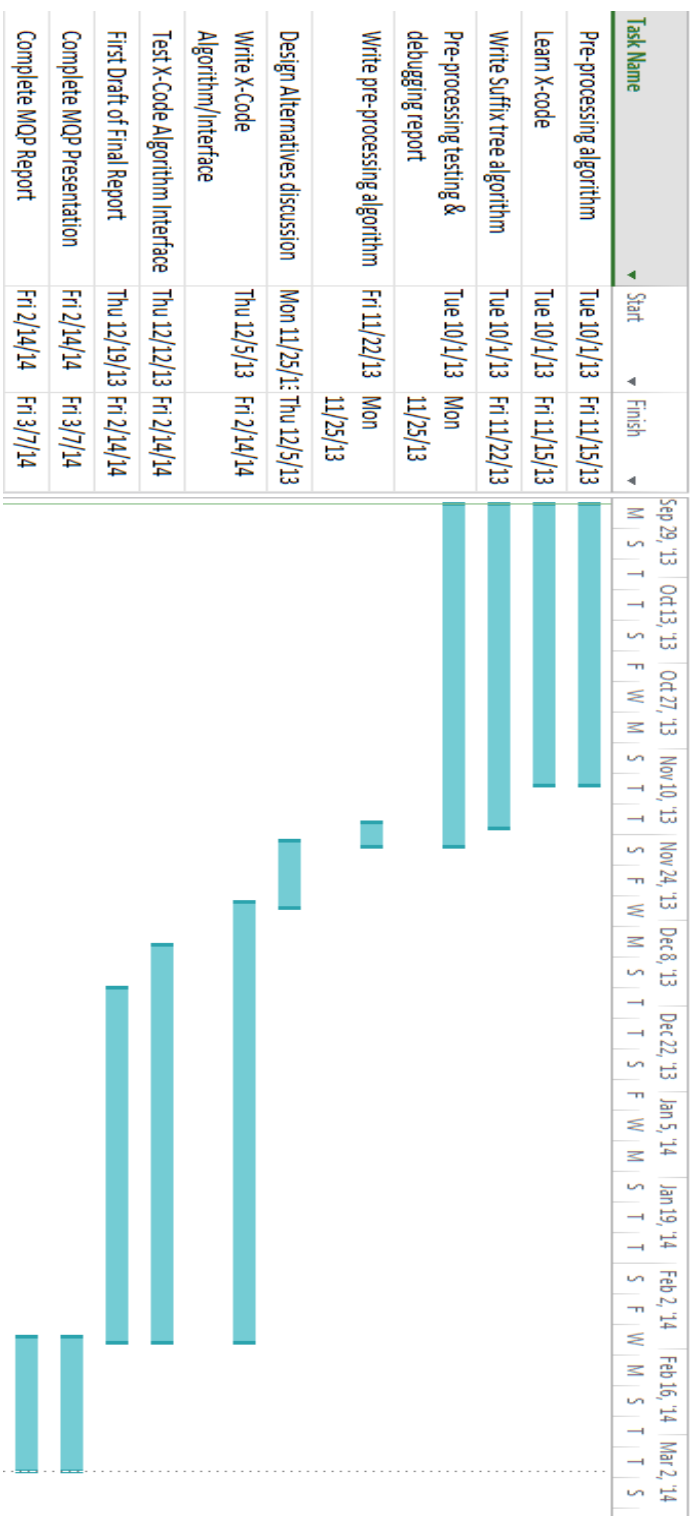
## References

1. National Institutes of Health, "An Overview of the Human Genome Project," 2012. [Online]. Available: <http://www.genome.gov/12011238>. [Accessed October 2013].
2. BRCA1 and BRCA2: Cancer Risk and Genetic Testing Fact Sheet - National Cancer Institute. (n.d.). *BRCA1 and BRCA2: Cancer Risk and Genetic Testing Fact Sheet - National Cancer Institute*. Retrieved , from <http://www.cancer.gov/cancertopics/factsheet/Risk/BRCA>
3. Campeau PM, Foulkes WD, Tischkowitz MD. Hereditary breast cancer: New genetic developments, new therapeutic avenues. *Human Genetics* 2008; 124(1):31–42.
4. Surgery to Reduce the Risk of Breast Cancer Fact Sheet - National Cancer Institute. (n.d.). *Surgery to Reduce the Risk of Breast Cancer Fact Sheet - National Cancer Institute*. Retrieved , from <http://www.cancer.gov/cancertopics/factsheet/Therapy/risk-reducing-surgery>
5. National Institutes of Health, "An Overview of the Human Genome Project," 2012. [Online]. Available: <http://www.genome.gov/12011238>.
6. National Institutes of Health, "A Brief Guide to Genomics," National Human Genome Research Institute, 2011. [Online]. Available: <http://www.genome.gov/18016863>.
7. Wetterstrand KA. DNA Sequencing Costs: Data from the NHGRI Genome Sequencing Program (GSP) Available at: [www.genome.gov/sequencingcosts](http://www.genome.gov/sequencingcosts).
8. Human Genome Project Completion: Frequently Asked Questions. (n.d.). *Human Genome Project Completion: Frequently Asked Questions*. Retrieved , from <http://www.genome.gov/11006943>
9. National Institutes of Health, "Genetic Mapping," National Human Genome Research Institute, 2012. [Online]. Available: <http://www.genome.gov/10000715>.
10. C. Maher, "Massively Parallel Sequencing," Washington University in St. Louis, 2012
11. Illumina: Liu, B. (2014, April 1). Telephone Interview
12. I. Mandoiu, and A. Zelikovsky, "Efficient Combinatorial Algorithms For DNA Sequence Processing," *Bioinformatics Algorithms : Techniques and Applications*, pp. 223-239, Hoboken, NJ: Wiley, 2008
13. Metzker, M., "Emerging technologies in DNA sequencing", *Genome Res.* 2005. 15: 1767-1776
14. Oxford Nanopore Technologies Ltd. (n.d.). The MinION device: a miniaturised sensing system. Retrieved April 30, 2014, from <https://www.nanoporetech.com/technology/the-minion-device-a-miniaturised-sensing-system/the-minion-device-a-miniaturised-sensing-system>
15. V. Bonazzi, "Genome Informatics," National Human Genome Research Institute, [Online]. Available:

[http://www.genome.gov/Pages/Newsroom/Webcasts/2010ScienceReportersWorkshop/Bonazzi\\_Informatics\\_VRB.pdf](http://www.genome.gov/Pages/Newsroom/Webcasts/2010ScienceReportersWorkshop/Bonazzi_Informatics_VRB.pdf)

16. National Institutes of Health, "Answers to Genome Analysis May Be in the Clouds," National Human Genome Research Institute, 2012. [Online]. Available: <http://www.genome.gov/27538886>.
17. Rocchi, C. (2011, June 28). Networking Tutorial for iOS: How To Create A Socket Based Server. Ray Wenderlich. Retrieved November 2013, from <http://www.raywenderlich.com/3932/networking-tutorial-for-ios-how-to-create-a-socket-based-iphone-app-and-server>
18. Zhao, M., & Lee, W. (2013, April 10). Complete-Striped-Smith-Waterman-Library. *GitHub*. Retrieved , from <https://github.com/mengyao/Complete-Striped-Smith-Waterman-Library>
19. Price Plans for Powerful iOS Controls | ShinobiControls. (n.d.). *Price Plans for Powerful iOS Controls / ShinobiControls*. Retrieved , from <http://www.shinobicontrols.com/ios/shinobisuite/price-plans>

## Appendix A – Gantt Chart



# Appendix B- Work Breakdown Structure

