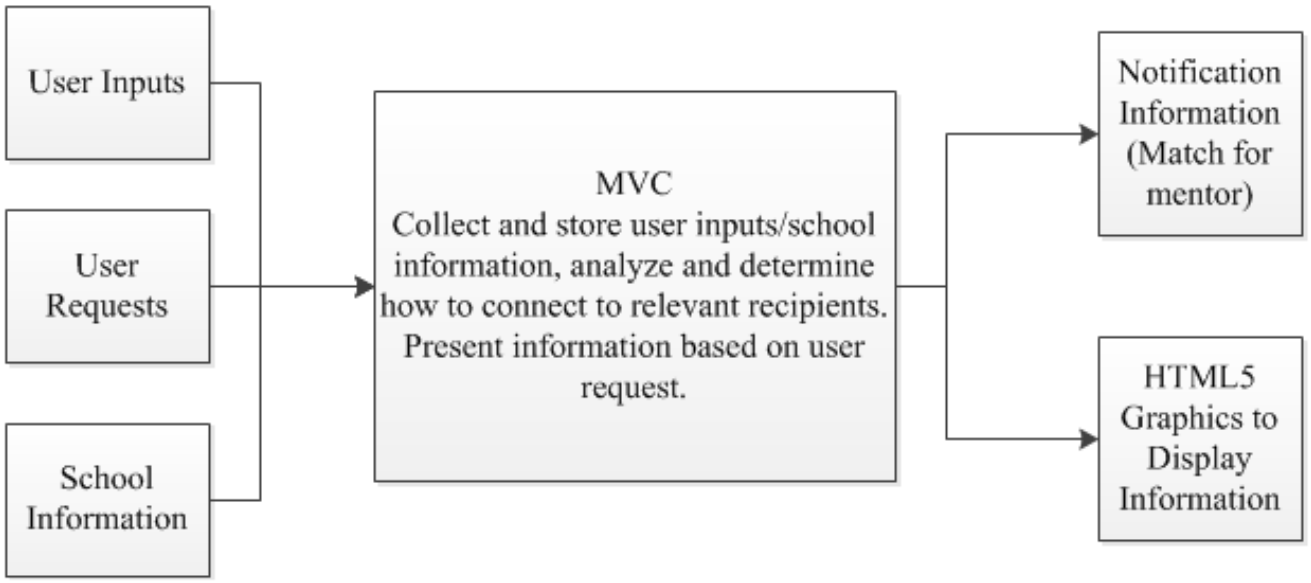
Section 1

Eric Cherin was responsible for creating the black box diagram. Robert Pinto designed the function-means tree diagram. Vincent Gasbarro elaborated on the technical explanations for both the black box and function-means tree diagrams.

	Vincent Gasbarro	Eric Cherin	Robert Pinto
Percentage of Effort	34%	33%	33%

Section 2

Black Box:



The black box defined by the group represents how inputs are handled server side of the application through the use of a model-view-controller (MVC) model. The goal of the web service is to connect high school students to college students and facilitate the passing of information from mentor to mentee. To accomplish this, vast amounts of information must be processed between involved parties. Thus the role of the server/MVC model is to aggregate information from the user base and relay it according to user requests. All user information is to be self-generated, but the inclusion of other information is still being debated by the group.

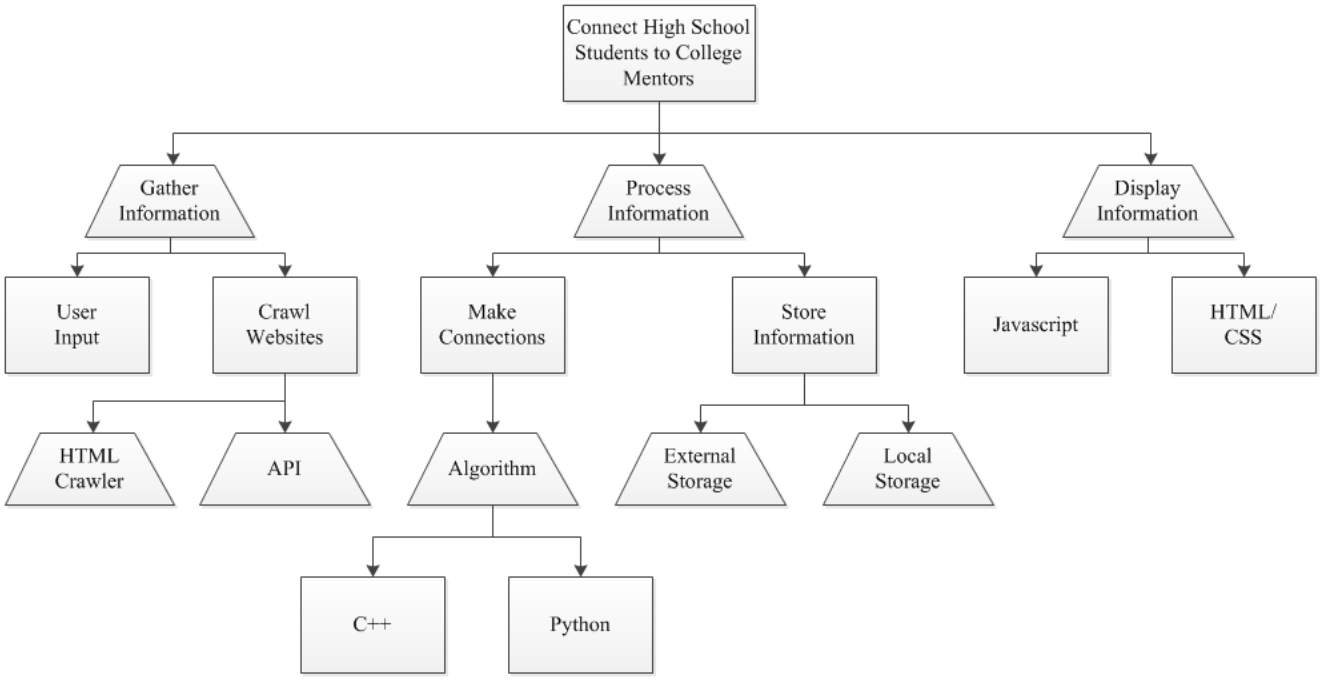
As the service provided is mentor based, user generated information will be the primary data input. User input refers to both mentees asking for help and mentors providing aid. This data must be collected by the server and processed. For example, user site information such as login must be saved to a database before being updated to the front end interface. Also, all user inputs must be processed before being matched to user requests. If a mentee is looking for information pertaining to a specific college or interest, the server must decide whether it has relevant information or must post a new request for a mentor. User requests are handled in the same manner, first being cross checked against existing informations before being fully disclosed to

the website. Examples of user data that would be collected through user inputs and requests include user preferences, personal details, and user queries. The server will need to analyze different tags in order to suggest appropriate mentor-mentee pairings. The exact algorithm for this process has yet to be decided.

The school information input refers to information obtained from school websites and personnel. This source of information is a very important data point for use in both user comparison and advertising. The group's web service is focused on making the transition from high school to college both easier and more intuitive for high school students. Thus information of colleges must be present on the web service to provide overviews of the schools and important metrics such as acceptance requirements and programs offered. To implement this information, there are two possible solutions. The first would be to read a school's website API and parse the important information automatically to the web service. The other option would be to ask school representatives who wish to use the service to provide relevant information in a form-like submission. From there, the web service could be designed to automatically update information as it changes.

The primary output of the server would be the organized display of the inputted information. The server would handle the formatting of all the information gathered, but without a visual representation of that data, there would be no value for end users. Thus, by using an industry standard such as HTML5 and CSS, the user would have a graphically friendly environment to work in. All the generated output depends on user inputs and must be continually updated based on user interactions. Thus, a secondary output of notifications is important to inform users when a pertinent event occurs. These notifications can vary depending on the event's level of importance. A simple pop-up notification would be sufficient for comments or postings on a user's profile. For larger events such as a matching of a mentor to mentee or updates to the application, an email would be sent to notify users.

Function-Means Tree:



The function-means tree analyzes the various functions and required means of execution in order to satisfy the needs of the system. For the group's web service, the three primary means to support the function of the system are to gather information, process information, and display information.

The gathering information mean obtains all the necessary input for the web service to operate and loosely represents the controller aspect of the MVC pattern. The two functions required are user input and crawling websites. As explained before through the black box, user input would be the greatest source of information. Further examples of user input include discussion topics, mentoring questions, and personal insights. The group will be able to facilitate user input though formatting a combination of HTML forms and a database. Crawling websites would be the second source of information, with two means of execution. A HTML crawler would allow for parsing information from websites automatically. This ability would be key in gathering comparative statistics from college websites and news articles. Obtaining information from news articles about subjects such as return on investment and nationwide rankings would be greatly beneficial for the decision making process. Social media pages might also be a useful source of information of college promotions and events. If a HTML crawler is insufficient, using the API of the target website would allow for the group's web service to obtain the necessary information. Further research must be conducted on what APIs are available for public use and how to actually integrate an API parser.

Processing the gathered information would be the second mean and loosely represents the model in the MVC pattern. The primary functions of this mean are to make connections based on

user inputs and storing the information being inputted into the web service. The making connections function is the entire premise of the group's web service. It is a broad function in the sense that it needs to be able to both match similar mentees with mentors and identify the relevance of user inputs. To accomplish these tasks, a high level programming language such as python or C++ can be used to design algorithms. At this present time, the group is unsure of what specific algorithmic details and techniques it intends to use but further research and testing are viable paths. Generally thinking, the group knows that a searching algorithm, a sorting algorithm, and a matching algorithm must all be designed to ensure proper workings of the web service. Another possible high level language that could be used is java, but the group is currently better versed with C++ and python, thus the exclusion of java as a function. With all the gathered information, the web service will need a function to store information. There are two means of accomplishing this, external storage and internal storage. External storage refers to making use of either networked data servers or the cloud. Both are viable options that in the end will be decided upon by their costs. At the beginning of the project, the group can make use of its personal networked storage devices to reduce cost. However, as the project scales, these personal storages could quickly run out of room thus requiring the use of commercially available data stores. The cheapest form of data stores would be renting space in the cloud. Amazon has a stabilized and cost effective cloud architecture in place that the group could use to complement the Amazon EC2 server that the web service could be running off of. Internal storage refers to storing the data within the system's application. While this method would lead to very high performance, internal storage is limited by disk space and could quickly fill up. To compromise between the two means, currently active data could be stored internally to improve performance while historical data could be saved externally as performance isn't as highly desirable.

The final mean required by the system is displaying the information, which represents the view of the MVC pattern. This mean will be the front end of the web service that all users will interact with. It will facilitate gathering information as users view currently available topics and decide on what information to contribute or request of the application. It will need to be both aesthetically pleasing and functionally enticing for the users. To accomplish these goals, this mean's two functions are JavaScript and HTML/CSS. These functions are current industry standards of web pages and have a plethora of documentation, tutorials, and support. The group is in the process of designing a basic landing page to serve as a minimum viable product and begin attracting user attention and feedback. Another benefit of HTML is that it is a global and cross platform standard. All browsers on both computers and mobile devices support HTML and by extension CSS, thus the developer's job will be much easier. In the future, the group might need to revise its template and choices of languages to allow for further integration as an application on mobile devices, but in its current state, the web service will be best used in browser format only.