Team First Light

Requirements Specification

Version 1.2

1 December 2021

Sponsors

Dr. David Trilling

Dr. Mike Gowanlock

Mentor

Felicity Escarzaga

The Team

Matt List- mjl79@nau.edu

Carson Pociask - cmp557@nau.edu

Jakob Nelson - jrn235@nau.edu

William Fuertes- wf69@nau.edu

Jensen Roe - jr2999@nau.edu

Accepted as baseline requirements for the project

For the clients: __________________________ *For the team:* __________________________

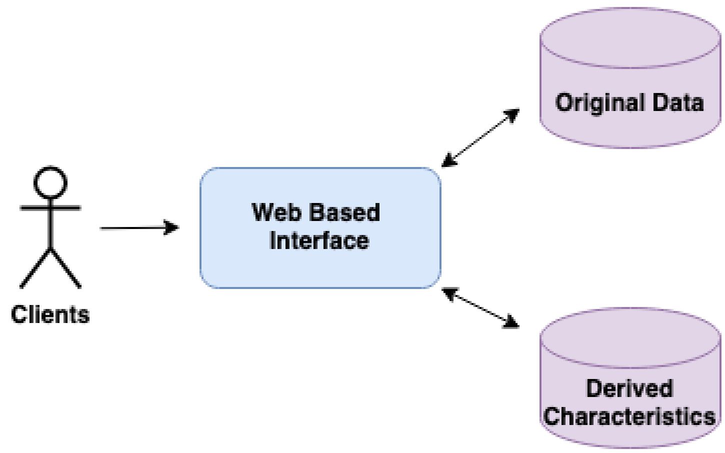
____

Table of Contents

1.0 Introduction

Big data is a relatively new field in technology. The process of extracting information from sets of data too massive to be handled by ordinary technologies can solve difficult business problems. Solving these problems are at the forefront of many of today's technological efforts, and while efficient tools already exist to send, capture, and store large sets of data, the need to perform analysis on them is growing. Forbes predicts that more than 150 zettabytes will need analysis by 2025. Researchers, businesses, and everyday people are responsible for today's massive growth in data creation. In order to fully utilize massive amounts of data, it is important that the technologies used to support its operations keep up with the demand of today's growth in data. While it may seem like most of the world's big data technologies are cutting edge, some fields still lack the support needed to draw useful information out of collected data. One of the fields that experiences drawbacks in workflow due to this lack of support is astronomy.

The field of astronomy is rapidly expanding and the data pertaining to night sky observations continues to grow in size, on the scale of terabytes and petabytes. Analyzing astronomical data is becoming increasingly complex and has become much more involved with data science as a whole. An example of this growth in today's world is the Vera C. Rubin Observatory. Currently being constructed on Cerro Pachón in Chile, this site, when operational, will produce roughly 20 terabytes of data every night under the 10-year Legacy Survey of Space and Time (LSST). The telescope will be taking all-sky observations, meaning that astronomical data will be compiled from all visible parts of the sky on a nightly basis. Different researchers across the field of astronomy will handle some percentage of this produced data, and Northern Arizona University (NAU) will be ingesting datasets pertaining to asteroids. The clients that will support this team are professors at NAU: Dr. David Trilling, Professor of Astronomy and Planetary Science, and Dr. Mike Gowanlock, Assistant Professor at the School of Informatics, Computing, and Cyber Systems (SICCS). Both clients perform analysis on the observational data pertaining to asteroids, or rocky bodies in the solar system that are tracers of the formation and evolution of the solar system. Since construction in Chile is still underway, a testbed is being used from the Zwicky Transient Facility (ZTF) located in San Diego, which is currently operating in a fashion similar to the LSST. Once arrived into NAU's database, both Dr. Trilling and Dr. Gowanlock utilize the captured data to draw out further information to perform research and simple observational work. **Figure 1.1** depicts a simple workflow of the clients where a web based interface uses data from two databases to represent to the user. The tracking of asteroids' characteristics over time is a common task for the clients and is useful for learning more about the specifics of the solar system. These observations and the research conducted on the collected data has the ability to help researchers further understand the history and current state of the solar system. Tracking asteroids as they travel across the night sky also helps predict the likelihood of asteroids coming in close contact with the earth.



`

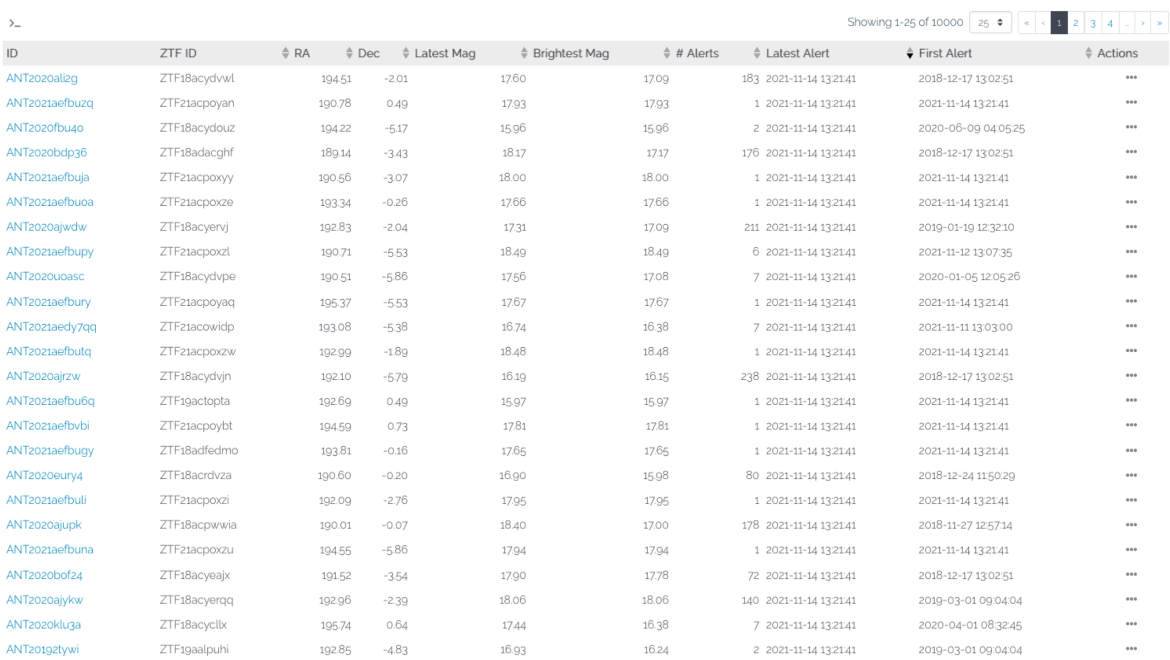
Figure 1.1 - Client Workflow

Both Dr. Trilling and Dr. Gowanlock interact with other researchers, typically on a local level, and often share information relating to astronomical observations. The workflow of the clients and other astronomers is driven by the need and desire to better understand the solar system. Research supported by tools like the one to be built by this team helps deepen an understanding of the solar system.

2.0 Problem Statement

As previously mentioned, there are already well-built tools in place that are able to capture and transfer very large sets of data. However, once astronomical data is gathered and stored, there exists a bottleneck. This restriction in workflow is caused by the need for data to be analyzed with the use of visualizations to provide more in-depth and easily accessible information. However, the current solutions are lacking in quantity and quality. The workflow of the clients deals with various characteristics derived from observed asteroids, such as magnitude or size. Comparing these characteristics over time using rudimentary graphical tools can prove difficult and problematic when very large data sets are to be considered, on the order of terabytes produced on a daily basis. Currently, utilizing this derived data during research is time consuming to use and difficult to understand due to the lack of graphical support. This greatly hinders the workflow of the clients and other astronomers; the data is collected yet an efficient way to utilize it for useful research is lacking. While the tools that the clients currently use do include visualizations of data, they are lackluster and will not be capable of handling the massive amounts of data that will eventually be used from the Vera C. Rubin Observatory.

A couple of example sites that the clients have previously dealt with encompass the main issues just discussed. The sites are ANTARES (antares.noirlab.edu/loci), and MARS (mars.lco.global/), and they are both centered around collecting and displaying massive astronomical data sets. The main list on the home page of ANTARES can be observed in **Figure 2.1**. A massive list of different asteroids can be observed here, where each has its own link to more in depth information. While the site is organized, the way that the large amount of information is displayed to the user is very overwhelming and uninviting. The list observed in **Figure 2.1** makes simple observations difficult, i.e determining outliers based on the asteroids' brightness. Also notice that no visualizations exist to support ease of use for this website. These setbacks in technology are the kinds of things that hinder both the clients' and other astronomical researchers' ability to conduct effective research.


>_ Showing 1-25 of 10000 25 « ‹ 1 2 3 4 .. › »

ID	ZTF ID	RA	Dec	Latest Mag	Brightest Mag	# Alerts	Latest Alert	First Alert	Actions
ANT2020ali2g	ZTF18acydvwl	194.51	-2.01	17.60	17.09	183	2021-11-14 13:21:41	2018-12-17 13:02:51	•••
ANT2021aefbuzq	ZTF21acpoyan	190.78	0.49	17.93	17.93	1	2021-11-14 13:21:41	2021-11-14 13:21:41	•••
ANT2020fbu4o	ZTF18acydouz	194.22	-5.17	15.96	15.96	2	2021-11-14 13:21:41	2020-06-09 04:05:25	•••
ANT2020bdp36	ZTF18adacghf	189.14	-3.43	18.17	17.17	176	2021-11-14 13:21:41	2018-12-17 13:02:51	•••
ANT2021aefbuja	ZTF21acpoxyy	190.56	-3.07	18.00	18.00	1	2021-11-14 13:21:41	2021-11-14 13:21:41	•••
ANT2021aefbuoa	ZTF21acpoxze	193.34	-0.26	17.66	17.66	1	2021-11-14 13:21:41	2021-11-14 13:21:41	•••
ANT2020ajwdw	ZTF18acyervj	192.83	-2.04	17.31	17.09	211	2021-11-14 13:21:41	2019-01-19 12:32:10	•••
ANT2021aefbupy	ZTF21acpoxzl	190.71	-5.53	18.49	18.49	6	2021-11-14 13:21:41	2021-11-12 13:07:35	•••
ANT2020uoasc	ZTF18acydvpe	190.51	-5.86	17.56	17.08	7	2021-11-14 13:21:41	2020-01-05 12:05:26	•••
ANT2021aefbury	ZTF21acpoyaq	195.37	-5.53	17.67	17.67	1	2021-11-14 13:21:41	2021-11-14 13:21:41	•••
ANT2021aedy7qq	ZTF21acowidp	193.08	-5.38	16.74	16.38	7	2021-11-14 13:21:41	2021-11-11 13:03:00	•••
ANT2021aefbutq	ZTF21acpoxzw	192.99	-1.89	18.48	18.48	1	2021-11-14 13:21:41	2021-11-14 13:21:41	•••
ANT2020ajrzw	ZTF18acydvjn	192.10	-5.79	16.19	16.15	238	2021-11-14 13:21:41	2018-12-17 13:02:51	•••
ANT2021aefbu6q	ZTF19actopta	192.69	0.49	15.97	15.97	1	2021-11-14 13:21:41	2021-11-14 13:21:41	•••
ANT2021aefbvbi	ZTF21acpoybt	194.59	0.73	17.81	17.81	1	2021-11-14 13:21:41	2021-11-14 13:21:41	•••
ANT2021aefbugy	ZTF18adfedmo	193.81	-0.16	17.65	17.65	1	2021-11-14 13:21:41	2021-11-14 13:21:41	•••
ANT2020eury4	ZTF18acrdvza	190.60	-0.20	16.90	15.98	80	2021-11-14 13:21:41	2018-12-24 11:50:29	•••
ANT2021aefbuli	ZTF21acpoxzi	192.09	-2.76	17.95	17.95	1	2021-11-14 13:21:41	2021-11-14 13:21:41	•••
ANT2020ajupk	ZTF18acpwwia	190.01	-0.07	18.40	17.00	178	2021-11-14 13:21:41	2018-11-27 12:57:14	•••
ANT2021aefbuna	ZTF21acpoxzu	194.55	-5.86	17.94	17.94	1	2021-11-14 13:21:41	2021-11-14 13:21:41	•••
ANT2020bof24	ZTF18acyeajx	191.52	-3.54	17.90	17.78	72	2021-11-14 13:21:41	2018-12-17 13:02:51	•••
ANT2020ajykw	ZTF18acyerqq	192.96	-2.39	18.06	18.06	140	2021-11-14 13:21:41	2019-03-01 09:04:04	•••
ANT2020klu3a	ZTF18acycllx	195.74	0.64	17.44	16.38	7	2021-11-14 13:21:41	2020-04-01 08:32:45	•••
ANT2019ztywi	ZTF19aalpuhi	192.85	-4.83	16.93	16.24	2	2021-11-14 13:21:41	2019-03-01 09:04:04	•••

Figure 2.1 - ANTARES Site Home Page

To bring all these problematic points together is helpful in defining a set of problems that the clients seek to have a solution to, and ANTARES acts as a good reference point. In a concise summary, the current problems that the clients and other astronomical researchers are experiencing are as follows:

- Lack of visual support, i.e plots, charts, and graphs of large data sets
- Lack of interactivity among large data sets
- Uninviting website pages, i.e massive lists of data points that are difficult to navigate

- The clients' current solution is unusable due to server hosting conflicts.

3.0 Solution Vision

Now that the problems that the clients and other researchers experience are layed out, a solution can be determined and expanded upon. The solution for this project is a web interface capable of supporting visualization and interactivity among cleanly represented astronomical data sets terabytes in size. This solution will serve as a simple and easy to use base for astronomical observation and research, specifically on asteroids. In further detail, the solution will encompass the following features to address the problems at hand:

- Visualization of large data sets through graphs, charts, plots, etc.
- Interactivity of data points within visualizations
- Simple web interface with easy to navigate web pages that pertain to different modules, i.e home page with large data summary vs. page with specifics of chosen asteroid

The ability to pull astronomical data from the database at NAU is crucial as said data is the center point of this project. While the computational operations such as deriving characteristics from asteroids is already complete, both the original and derived data still need to be retrieved from the database in order to continue onto visualization and interactivity. Once the data is retrieved, storing the large sets in memory will need to be considered. One of the most important aspects of the envisioned solution is creating graphical representations on a web page out of the large data sets from tabular structures. This operation will address one of the most crucial problems of the clients, a lack of data visualization. Having data in graphical formats rather than long lists will help ease the research and observation process for the clients and alike researchers. **Figure 3.1** represents a simple, yet improved workflow of the clients.

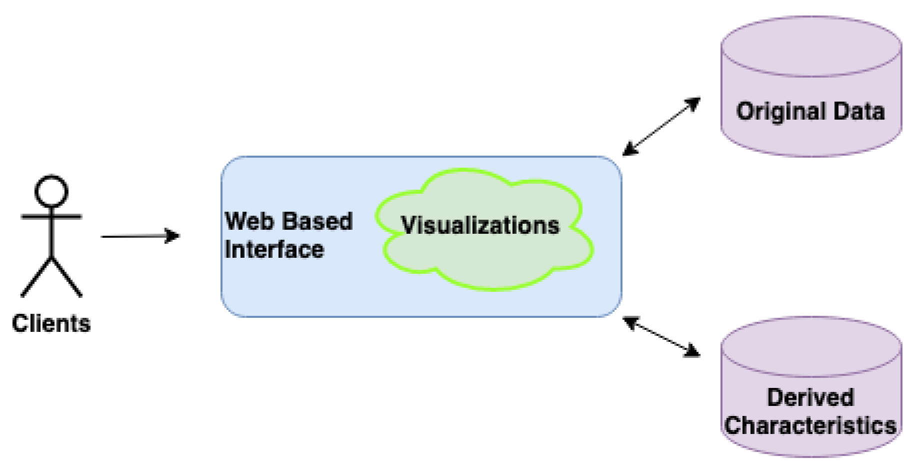


Figure 3.1 - Improved Client Workflow

More effort can be devoted to conducting useful research on the data rather than taking time to navigate through data points that are difficult to tell apart or group together. **Figure 3.2** does an excellent job representing datasets in a meaningful way and observations such as determining outliers are easy to do. Relative to **Figure 2.1** that was previously discussed, the data in **Figure 3.2** is easily summarized in a way that is inviting and easy to observe.

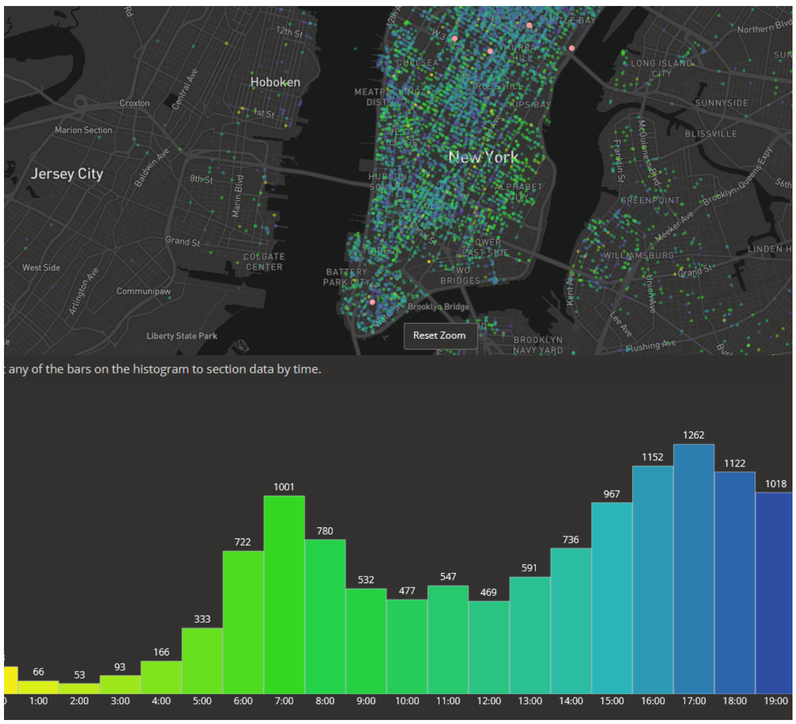


Figure 3.2 - Uber Data Visualization Example

For this project, not many other solutions were considered. The problems to be solved are well-defined and are often solved in similar ways across different fields. While the clients and other astronomical researchers that will be using this solution make up a relatively small group, maybe on the order of hundreds of people, there is still a chance it makes a broader impact in the field of astronomy. Easier and more time-efficient experiences during research and observation could inspire further, more in depth research that could entail a much larger audience.

4.0 Project Requirements

A well established problem statement and solution vision allows for an in depth look at this project's requirements. To start, the domain-level requirements must be established. This will allow for more specific and tailored requirements to be explained in subsequent sections to ensure full coverage and understanding of what the clients and the team envision as a final product. The domain-level requirements defined for this project are as follows:

- **Visualizations of large percentages of the observations in the database** - A high-level representation of the database should be readily available to users.
- **Interactivity among large data visualizations** - Data points should be able to be selected out of a large representation of data.
- **Ease of use** - Product should provide ease of use when navigating different data on web pages. Product needs to be simplistic in nature so as to not overwhelm users.
- **Filtering the data** - Datasets should be able to be filtered and visualized on the basis of different characteristics such as magnitude.
- **Tracking asteroids over time** - Asteroids should have the ability to be saved by a user to serve as a consistent reference point.
- **Widely accessible** - Product needs to be web-based to provide access to anyone who wishes to view and/or interact with the data. This also expands the reach to those who did not have access to the data beforehand.
- **Up-to-date data** - Product needs to display the most current observations possible as new astronomical data is to be generated on a daily basis.

Now that the domain-level requirements have been laid out, each can be further evaluated and explained. The following subsections entail the functional, non-functional, and environmental requirements. Each of which may branch into further requirements.

4.1 Functional Requirements

Some of the most important aspects for a successful delivery of this project revolve around the functionality the solution will support. The following section will look into the functional requirements for this project in a hierarchical manner beginning with high level requirements, thus providing the ability for other, more specific requirements to be branched off and explained in detail. As an overview, the functional requirements that will be looked into are as follows:

- Simplistic dashboard design to minimize clutter and improve navigation and visualization
- Visualize millions of objects
- Ability to interact with individual asteroids
- Ability to save/track specific asteroids
- Account creation and access in order to store asteroids for users

4.1.1 Simplistic Dashboard

This project will operate around a web application shared by potentially hundreds of astronomical researchers. A dashboard will serve as the center point in this web application as to provide functionality among the astronomical data. This requirement is important to complete successfully as it is an aspect that is commonly done poorly in current solutions used by the clients and other astronomical researchers. Referring back to **Figure 2.1**, the overwhelming amounts of data displayed is not appealing to interact with. The dashboard to be created needs to be simplistic in structure and style. The dashboard should also be organized, with links to other parts of the website available so that the main home page does not get cluttered. The sub-requirements of a simplistic dashboard include the ability to save and track asteroids through account creation and modification, access to downloadable data, and access to pages through links on the home page. These are further described below:

1. **Account Creation**
   An important sub requirement is that users should be able to have an account within the web application. Users will be able to create an account using a chosen username and password. An example of a feature for an account is the ability to link an email address and receive notifications of changes to the website's data. Note that this is an example and is not guaranteed to be implemented in the final solution for this project.

2. **Saving and Tracking Asteroids**
   The ability to reference back to asteroids supports ease of use in the clients' and other researchers' workflow. A user of the solution to be built should be able to save specific asteroids and easily access them through their account. This will act as a bookmark that the user can use on asteroids of their choosing. As understood from the clients, research will vary in astronomy, even among the asteroids that will be utilized in this project. This means that different researchers will want to utilize the solution for their specific purpose(s). This requirement makes it possible for a user to bypass functionality and web pages that do not contain what they are looking for.

3. **Downloadable Data**
   The ability to download data in various formats is another sub requirement. The user of the solution should be able to download various file formats pertaining to chosen

asteroids. In example, but not guaranteed in a final solution, a user will be able to download a PDF or PNG file of a scatterplot pertaining to an asteroid.

4. **Account Update**
   Modifications to accounts should be made easily accessible and take immediate effect. Deletion of saved asteroids is an example of an account modification.

5. **Account Login**
   A user should be able to login with their previously created credentials. A sign in page will be made available on the home screen of the web application.

6. **Forgotten Password**
   Should a user forget their password at account login, a method such as email will be used to support password recovery and/or reset. Note that this is an example and is not guaranteed to be implemented in the final solution for this project.

7. **Links to other Pages**
   On the homepage of the web application, there needs to be links to other pages. These pages will host modules such as account information or database information. Note that these modules may vary.

4.1.2 Data Visualization

The main aspect of this project's solution is visualizing millions of asteroids. Data visualizations will change how astronomical researchers view asteroids and support their research. As mentioned before, astronomical researchers have very few ways to quickly look at asteroid data and come up with inferences based on that said data without manually checking each individual asteroid and the data that is associated with it. Without the visualizations, research surrounding asteroids is slow and tiresome. Various visualizations will be created out of the astronomical data. Some sub requirements of the data visualization requirement include interactivity among various visualizations, a heatmap that groups asteroids into regions, as well as a scatter plot that will then be used to chart each asteroid in a specific region. Below, the sub requirements of data visualization are described.

1. **Interactivity Among Datasets**
   Visualizations are very important to this project and provide for a great point of data reference. However, interactivity within the visualizations expands the reach and overall functionality of the product, so it will also be implemented as a sub requirement of data visualization. The ability to interact with data points also allows for more effective in-depth research and observations to take place. Interacting with data points within a

visualization provides for better ease of use as described in a workflow scenario example below.

Scenario in Workflow : User is at the home page of the web application where a snapshot of the database is displayed in a visual. While observing, the user notices an asteroid that has unusually high magnitude and is interested in learning more about that particular asteroid. Instead of digging through thousands of data points in an attempt to locate the particular asteroid, the user simply selects the data point off of the visual. This will save a lot of time in the workflow as less time is spent manually searching extensive, uninviting lists of asteroids.

Not only does interactivity among visuals improve ease of use in a workflow, it will also create a welcoming environment for the product. An easy to use solution supported by interactivity among visualizations can help draw in more researchers to NAU who wish to use the created tools.

2. **Heatmap Visualization**
   A sub requirement of overall data visualization is the creation and use of heatmaps. A heatmap represents data using color codes and can differentiate asteroid characteristics quite easily. For example, a heatmap can serve as a high level view of the database, yet this is not necessarily guaranteed in the final solution.

3. **Scatter Plot Visualization**
   Another sub requirement of data visualization, similar to the previous, is the creation and use of scatter plots. Scatter plots are another form of visualization that will be used in this project, potentially on individual asteroids, serving as a representation of characteristics over time.

4.2 Non-functional (Performance) Requirements

In the context of this project, non-functional requirements play an important role in the overall success and usability of the final product. The amount of data handled during development is large enough that ordinary data tools are not feasible. When dealing with millions of data points, performance is often a bottleneck in the workflow of the user. Thus, it is very important to consider and thoroughly explain the non-functional requirements for this project. These requirements are listed below and will be evaluated throughout this section.

- Ability to visualize datasets within a reasonable amount of time
- Ability to handle multiple users making visualization requests simultaneously

4.2.1 Efficient Dataset Visualization

The database currently contains terabytes of data, and the solution will need to create multiple graphs, some featuring multiple terabytes of data. A page that loads slowly will frustrate users, which will decrease the usability and desirability of the solution page. The final solution will need to have answers for making the graphs available quickly, such as building and storing the graphs that will require the largest amount of data once a day, and only building graphs that use a subset of the data as a user requests it. This will allow the most demanding graphs to be built when users are not using the website, but not requiring more data to store all possible graphs that a user may require. Because this will be dependent on how quickly the application can receive data from the database, the team will require a smaller graph to be drawn in less than one second. Additionally, the application needs to draw the larger graphs, such as the summary heatmaps, only once and cache the data within. That said, this should not be an expensive task. The team will require the application to draw the larger summary graphs in less than 10 seconds.

4.2.2 Ability to Handle Multiple Users

Because this application will be publicly available, it is conceivable that multiple users will access this site at the same time. While a large number of users at once is not expected, as this site will be primarily used by a niche audience, the team needs to be aware of the cost of running multiple users, and to do so without significant loss of performance. The team will test this site with five users accessing the site at once, and will look to make sure that there is no significant penalty to generating data visualizations. When testing with five users at once, the team will consider a slowdown of less than 10% for drawing individual visualizations to be acceptable.

4.3 Environmental Requirements

It is important to consider the context of the project at hand along with the preferences of the clients when determining requirements. In order for the final product to be put to good use and expanded upon, it must take environmental requirements into account. This will shape the project in a way so that the clients are knowledgeable and comfortable working with the final product. Listed below are the environmental requirements for this project.

- Accessible by anyone
- Website must be hosted on servers at NAU
- Website must interact with database at NAU
    - Database interactions must use the Structured Query Language, or SQL
    - Interactive graphs must be made with the data gathered from the database.

4.3.1 Accessible to Anyone

The sponsors want this application to be built in order to make the datasets they have built accessible to other researchers outside of NAU. Therefore the team will need to host this site on a publicly available platform. During requirements acquisition, the sponsors let the team know of a previous project that was hosted on AWS. While this provided the necessary access to the public, it also created unforeseen issues. The sponsors have instead suggested that the solution be hosted on NAU's public web server.

4.3.2 Hosting Website at NAU

One of the key environmental requirements of the project is that the website is to be hosted on NAU servers as preference of both clients. NAU provides access to dedicated servers free of cost and allows for direct access to the NAU database. Upon researching hosting on NAU's servers, this appears to be a good solution. Since the database is already hosted at NAU, it will be much easier to transfer the data from the database to the web server, rather than needing to transfer terabytes of data to AWS daily.

Once the project is completed, the clients will have the capability to share it amongst their research colleagues. With NAU's server, creation of the webpage will be easy to maintain since it is local to the University and provides the clients ease of access due to the fact that they both are faculty of NAU.

4.3.3 Interacting with NAU Database

In order for this project to be successful, the solution will need to have a way to interact with the database in order to pull information from it. This data will provide a way to create interactive graphs so that users can see more information on the thousands of data points. The database to be used in this project is SQL based and is hosted at NAU. This will give the team the ability to link the incoming data to the website.

1. **Using SQL**
   On a smaller side note, SQL will have to be used for database interactions due to the fact that the database is in SQL. This smaller requirement is nothing to worry about as plenty of tools that vary in functionality that are able to understand SQL already exist. Overall, this provides simplicity for the project during development and during maintenance/upgrading.

5.0 Potential Risks

The potential risks for this project are relatively low compared to what another capstone project could be experiencing. This is because the backend of this project is taken care of by the clients. The team will be using the NAU servers to host the web application, and all the data from the database is secured by NAU. Potentials risks identified for this project as as listed:

- Data Format Shift
- Security of the Web Application
- Packages Become Unsupported or Change
- Overload of Incoming Data
- Too Large of a User Pull Request
- Too Many User Pull Requests

5.1 Potential Risk Descriptions, Likelihood, and Severity

5.1.1 Potential Risk #1 - Data Format Shift

One potential risk is that the format of the data inside the database might shift. What this means is that there could be another characteristic column added for an asteroid, or that the numbers retrieved might only be input up to 2 decimal places after the dot placement instead of what it currently is input for. This could be a potential risk because the application would not support pulling the new column data into the web application to be visualized and graphed along with all of the other data that will already be visualized and interactive for users to view and study.

This is likely to happen at some point in the future during the continued development of the NAU database and as more and more data is gathered and derived. The severity of this risk is not as impactful as other risks with this project. While the web application would not support this risk if it happened, it is easily mitigated. The continued development and advancement of this project is something that will be expected to help support NAU being known as a leader in Astronomy and Planetary Sciences.

5.1.2 Potential Risk #2 - Security of the Web Application

Another risk is the security of the developed web application. If the application is not properly secured, such as hashing and salting passwords, and a user uses the same credentials elsewhere, the web application can be a risk to users. This team considers the security of users data to be important and is aware of the difficulty of protecting user data. While it is impossible for a public facing application to be perfectly secure, there are methods and best practices that the team can follow when building the web application that will best protect user data.

The risk of attack is probable. Every day, attackers scan for low hanging vulnerabilities. However, most of these attacks are basic and will not target the software stack used on this application, such as attacking a PHP based application. However, the team will need to make sure to follow best practices to protect usernames and passwords, such as hashing and salting the stored passwords. This way, even if the website is attacked, user data would be protected for as long as the hash and salt algorithm is secure.

5.1.3 Potential Risk #3 - Packages become Unsupported or Change

A third risk is that the packages used for this project become unsupported or change to a point where it is unusable to perform the desired functionality for this project. Software packages are frequently updated, and sometimes the new update phases out a previous part of functionality due to either it being relatively unused, better and more used options exist otherwise, or a new functionality is added that replaces the previous functionality. Even worse, the package the team used for this project is phased out completely for an unknown reason(s). This would be the worse-case scenario for this risk because the whole project would have to undergo a renovation and remodel. Overall, this is a risk because it would not allow full functionality for the project without maintenance and an update, and therefore not work to the full specifications outlined in this document.

This risk is neither likely nor unlikely to happen as the team and users do not know whether or not a developer is planning on changing a particular software package or not supporting it in the near future. However, this problem can be mitigated by locking in versions, and testing updates on a separate branch or installation to ensure that an update does not break the functionality of the application. Additionally, if an update would break some functionality, such as by an API change or depreciation, this would be a good opportunity for the developers to adjust or refactor the code base.

5.1.4 Potential Risk #4 - Overload of Incoming Data

Another risk would be if there was an overload of incoming data. This would be due to the web application being unable to hold an overt amount of data inputted into the system, like visualization construct constraints or a bug with the pull request from the NAU database. Another reason this risk might happen would be due to a change in the input data from the NAU database that is unsupported through the parameters given to the data visualizations.

This risk is likely to happen at some point largely in part to the massive amount of users that this project is expected to be available to. This would not be a severe risk as it could be mitigated as long as the issue is identified so that a solution can be figured out and implemented.

5.1.5 Potential Risk #5 - Too Large of a User Pull Request

To expand upon the previous risk of an overload of incoming data, too large of a user pull request is a risk to this project as well. Pull requests that request an amount of data that is too large for the pull request to perform is something that would be caused by the software that is used to pull the data from the NAU database. Too large of a pull request could lead to freezing or lagging in the performance of the web application that would have to utilize all of its resources to make the pull request.

This risk is likely to happen due to the research being performed on any number of asteroids that the NAU database will hold data for. It would be a semi-severe risk as it would lead to poor performance from the web application that could frustrate the users and researchers.

5.1.6 Potential Risk #6 - Too Many User Pull Requests

Another risk that could happen that relates to pull requests is if a user requests too many pull requests before one is completed. It is important that each pull request from a user is completed before another one is requested. This is to protect an overload of data the web application could experience, which would affect its ability to run smoothly and provide clean functionality for users.

This risk is unlikely to happen as users will want to view their pull requests and the visualizations that come along with the data pulled from the NAU database. The risk is not a severe issue if it happens, as steps will be taken to mitigate the risk.

5.1.7 Potential Risk #7 - Poor Performance with the Data Visualizations

The last risk that this project has is poor performance when interacting with the data visualizations. This risk will be the most common that the team encounters while developing the solution for this project. This is because the many terabytes of data that this project will be visualizing has a strong chance to slow down the website in terms of interactivity and performance. When interacting with a visualization, loading speeds of the sections or particular asteroids have to be able to be loaded in a set amount of time. If the web application that is being developed has slow performance in any part of the solution, then the sponsors and NAU will look poor in the eyes of users that will become frustrated with the slow speeds and inability to perform research in a timely manner.

This risk is very likely to happen and is severe enough to warrant immediate action by the team to make sure that it is mitigated as best as possible.

5.2 Potential Risk Mitigations

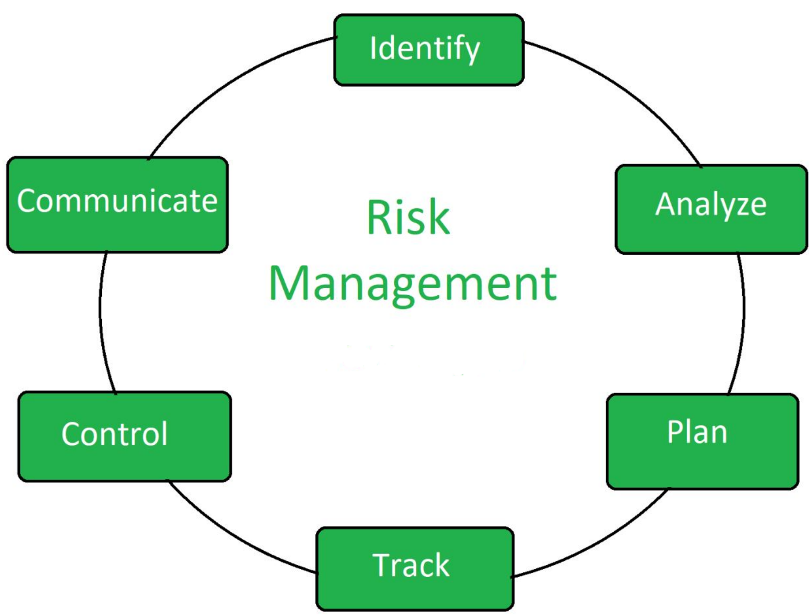


Figure 5.2.1 - Risk Management Technique

When encountering a risk that happens to occur during the development of this project, the team will refer to the techniques in **Figure 5.2.1** to manage the risk and find a solution.

5.2.1 Potential Risk #1 - Data Format Shift

The solution for this risk is to provide a way for the code to be easily changed to support another data format column as part of the data visualization and the respective data to be presented when an asteroid is interacted with by a user.

5.2.2 Potential Risk #2 - Security of Web Application

The solution to this problem is twofold. First, the team needs to make sure to use updated packages, and utilize good coding practices. This will limit the amount of vulnerable code for attackers to take advantage of. Additionally, any user passwords will need to be stored securely, such as comparing a login to a stored hash. This means in the event that the web application is breached, user data would still be protected.

5.2.3 Potential Risk #3 - Packages become Unsupported or Change

The solution to this risk is to provide a way for continued software development of this project through maintenance and extensive documentation for each and every part of functionality this project provides so that in the future if this risk happens, it can be easily mitigated.

5.2.4 Potential Risk #4 - Overload of Incoming Data

The solution to this risk is to make sure that the web application can optimally handle the way data is stored locally after it is pulled from the NAU database. This will be achieved by limiting the amount of data each pull request can pull. This ensures that the web application will not be overloaded by incoming data and can perform up to the standards that this project will be held to.

5.2.5 Potential Risk #5 - Too Large of a User Pull Request

Just like potential risk #4's solution, the solution to this risk is to make sure each pull request has a limited amount of data it can pull at one time. This denies the likelihood that this risk would happen.

5.2.6 Potential Risk #6 - Too Many User Pull Requests

The solution for this risk is to ensure that each pull request has been completed and reviewed before a new pull request can be requested. This alleviates the potential for too many user pull requests since one would have to finish and be reviewed by the user before the user can request another pull.

5.2.7 Potential Risk #7 - Poor Performance with the Data Visualizations

The solution for this risk is to ensure that the code supporting the web application will be able to handle the terabytes of data that will be queried in the NAU database and ready to be explored in a visualization. This will be done through extensive testing and optimization for parts of the code that will typically run slower than other parts due to the amount of data to be handled for a certain part of the project.

6.0 Project Plan

Now that the project problem, solution, and specifics are well defined, it is important that they are all tied together and put into a plan that is feasible to execute successfully. Currently, the team is busy with documentation and wrapping up the initial requirements phase as a whole. During the last month of 2021 and the entirety of the following Spring semester, the team will be working on developing and stitching together all the specified requirements to produce an effective end product. To further elaborate, a list of milestones can be found below accompanied by a rough date outline for completion, followed by descriptions of each.

- Host web application base at NAU and made simple interactions with the database. November 19th, 2021 - End of 2021.
- Begin to store massive amounts of data in local memory. November 19, 2021 - End of 2021.
- Create visualizations out of collected data. Being November 19th, 2021. This is an evolving process that will continue well into the Spring of 2022.
- Create interactivity among the visualizations. January 2022 - April 2022.
- Implement account creation and asteroid saving/tracking. January 2022 - April 2022.

These milestones will help guide seamless development as they build into each other and continue to evolve. The first step of hosting a basic web application on NAU's servers will provide a jumping off point for all the other functionalities. Alongside this operation will be the initial connection to the database. These two operations are requirements to begin development on this project in the Spring of 2022. From here, the team has a focused solution area to which additions can be developed as time progresses. **Figure 6.1** displays the workflow of the team from September 2021 through December 2021. This time frame mostly entails documentation, hence the lack of developmental phases. Coming up at the end of November and beginning of December, the team will be polishing off requirements and giving a presentation for the first design review. To conclude both the academic semester and the year, light development will take place in order to present content for the tech demo. These small development phases will also provide for a great jumping off point come the beginning of 2022, when the bulk of development will take place.

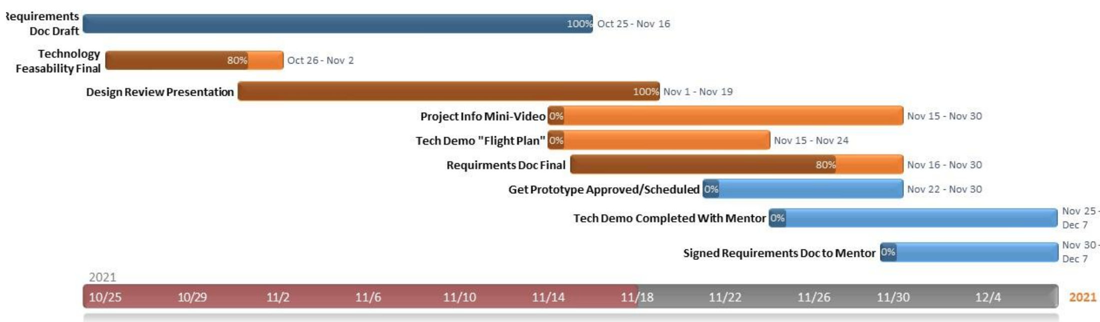


Figure 6.1 - Gantt Chart Outlining Workflow through December 2021

7.0 Conclusion

The visualization of astronomical data is a crucial component to analyzing asteroids, stars, and other objects. Without visualization tools for analysis, researchers need to comb through thousands of data points by hand, making the ability to see patterns and develop trends complicated. This project is intended to help researchers visualize and analyze the large amounts of data hosted in NAU's servers. The current methodology researchers use involves time intensive analysis that focuses on individual data points rather than larger trends and outliers. The solution outlined in this document is to create a web application that can pull large amounts of data and visualize it in graphs, heat maps, and other visualizations in order to help researchers understand the data they are looking at. The web application will give insight to outliers, trends, and distributions that will provide better understanding of what the data means. The functional and non-functional requirements outlined in this document provide not only the technical requirements that the web application must meet but also an outline of the roadmap the team intends to follow. The Gantt Chart in **Figure 6.1** shows the team roadmap as we near the end of the semester with a technical demo of the technologies chosen being presented in December. The team is confident that the scope of the project is manageable and that the technical demonstration in December will showcase the foundation of the project. The team is also excited that by May of 2022, researchers will have a new tool capable of visualizing vast amounts of astronomical data.