1. **System Requirements**
   1. **Programming Language:** The application is coded entirely in Python 3.
   2. **Capacity:** This application is 146MB.
   3. **Hardware:** System needs a minimum of 1 GB of RAM.
   4. **Software Dependencies:** System needs a minimum of Windows 7, MacOS 11, or Linux to run this. The application runs through a Python 3 interpreter. The instructions to set this up with the required packages and libraries are provided in the user manual.
   5. **External Dependencies:** This program is self-contained on the user’s computer. Future patches will be available on GitHub as they are developed. They will contain expanded constellations and information on them. The size of these patches and the software itself fits within GitHub’s free tier. Therefore users can download it straight from GitHub and no external server or database service is needed, at least for the initial version of this project.
2. **User Manual**
   1. **Software Setup**
      1. **Install Python**
         1. If Python 3 is already installed on your computer, skip to step 2.1.2.
         2. Otherwise, click the following link and click the ‘Download

Python 3.[current version]’ button shown below: <https://www.python.org/downloads/> .

A screenshot of a computer

Description automatically generated with medium confidence

* + - 1. Once downloaded, run the installer, and choose the selection for Python to set the PATH variable for you.
    1. **Download Software from GitHub**
       1. Next, click the following link to open the GitHub repository where the software package is stored: <https://github.com/jclanier11/SDEV265Group5Project>.
       2. Click the ‘Code’ drop down menu (step 1 below), then click ‘Download Zip’ (step 2 below).

A screenshot of a computer

Description automatically generated with medium confidence

* + 1. **Extract the Files**
       1. Open your computer’s downloads folder and extract the files from the ‘SDEV265Group5Project-main.zip’ zip file.
       2. After the process is complete, in the window that comes up (the new, extracted folder), open the single folder in it to access the directory with all the .py files in it. Leave that window open.
    2. **Run ‘pip install -r requirements.txt’ on Console**
       1. Using your console of choice, navigate it to the directory described in step 3.1.3.2 where all the .py files are, and run the command ‘pip install -r requirements.txt’ to install the Python libraries needed to run the program.
       2. If you are not familiar with how to navigate directories in the console, read this article for using [Mac’s Terminal console](https://www.macworld.com/article/221277/command-line-navigating-files-folders-mac-terminal.html) or this video for [Window’s Command Prompt](https://www.lifewire.com/change-directories-in-command-prompt-5185508#toc-how-do-i-change-directories-in-command-prompt).
          1. Once the last directory level to the left of the ‘>’ is ‘SDEV265Group5Project-main\SDEV265Group5Project-main’, type ‘pip install -r requirements.txt’ and hit enter. The console should install all the necessary libraries Python needs to run the software.

A screenshot of a computer

Description automatically generated

* 1. **Using the Program**
     1. **Open parameters.py**
        1. Reopen the file directory for the package with the .py files in it and click into parameters.py.

A screenshot of a computer

Description automatically generated with medium confidence

* + 1. **Select Parameters**
       1. Select the options from the drop down menus. Once selected, click the “Create Star Chart” button and the system will generate a set of results based on the chosen parameters.

A screenshot of a computer

Description automatically generated with medium confidence

* + - 1. Once you click the “Create Star Chart” button a loading status message will appear at the bottom of the window to inform you that the program is working on your request.

A screenshot of a computer

Description automatically generated with medium confidence

* + 1. **Star Chart**
       1. The system will generate the first of two windows, and it will display the star chart correlating to the given parameters in the first window. This star chart shows a map of the sky and all the constellations able to be found (see ‘1’ below). It maps out the constellations currently visible using lines and dots for easier readability (the larger the dot, the brighter the star). In addition, the following buttons extend the functionality of the Star Chart:

A screenshot of a computer screen

Description automatically generated with low confidence

* + - 1. ***Bottom Left Buttons and Their Functions***
         1. **House Button** - This button resets the view to what it was when the window was first generated.
         2. **Left and Right Arrow Buttons** - Similar functionality to undo/redo, these buttons allow you to go back to the previous view you chose with the zoom or pan options (left arrow), or go to the next view in your view history if you previously went backwards in it (right arrow).
         3. **Arrowed Cross Button** - The left mouse button allows the user to click and drag the image, panning the view. The right mouse button zooms the view out when clicking and dragging either left or down and zooms in when right clicking and dragging either right or up . Holding down CTRL and dragging in any direction causes the circle view to shift diagonally up and right or left and down, depending on which direction is closer to where the user is dragging it.

*While this is a helpful feature, because of the program configuration, it really only provides functionality if you have zoomed into the chart already using the magnifying glass button. Otherwise it just moves the sky view circle around the window.*

* + - * 1. **Magnifying Glass Button** - This button allows you to click and drag to select a desired view section that will become the new view, filling the window with that selection area.
        2. **Setting Sliders Button** - This feature comes as a part of an internal program component, and does not provide helpful functionality to this specific program.
        3. **Floppy Disk Button** - This feature saves the figure as a .png file to the desired location.
    1. **Constellations List**
       1. The second of the two windows to generate displays a list of the Zodiac constellations (more are planned to be added in later updates) that are best viewed in the time window selected by the parameters given in the first window. Each button will take you to the respective constellation information window *(see 3.2.5)*.

A screenshot of a computer screen

Description automatically generated with medium confidence

* + 1. **Individual Constellation Information**
       1. This page displays an image to help you find and map out the specific constellation you selected in the previous window with the list of constellations. *(see step 3.2.4)*. Along with the image, it also populates more information about that constellation.

A map of the constellations

Description automatically generated with low confidence

1. **User Requirements**
   1. This section outlines how our team was able to satisfy the requirements set forth for this project.
      1. The initial parameter window submit button causes a function to evaluate the parameters, then pulls from a Python class to display constellations in the Constellation List window that match those parameters.
      2. The Star Chart window interacts with an API to produce a picture of the night sky with constellations in it. This will display at the same time as the Constellation List window.
      3. The buttons on the Constellation List window will be clickable areas that generate the Information window. This window displays more in depth information about the selected constellation. The extra information is pulled from the same Python class mentioned in 4.2 above.