# Post Mortem

Goals that we have achieved:

* Setup typescript configuration for repo
* Setup Mocha test suite and hello world test
* Configure Docker environment
* Setup Bless terminal UI that displays "I am running!"
* Configure NPM script for building, running, and testing in one step

All of the goals have been met. No adjustments will be made to our project scope and capacity, since our A1 results have shown that our plan is working well so far.

# List of Features

A description of the features your project intends to implement (or next steps that you need to pursue) with sub-tasks prioritized

1. [Complete] Setup
2. [Complete] Backend Processing Class: this feature includes displaying text onto the terminal window from HTML, disregarding any formatting. We will take an URL or local file path from the command line, fetch the HTML data, then extract the plaintext, and finally output the plaintext onto the Blessed terminal window.

We are choosing to develop the tasks sequentially because rigorously defining multiple output structures at this early stage adds overhead that exponentially increases if any requirements to the output structure change.

* 1. [Myles] Setup simple dummy HTTP server to serve test pages and create some sample HTML pages. Also setup local file system to return HTML content  
       
     Priority: We’re prioritizing this first since most of the other features in milestone 2 will rely on the dummy HTTP server for testing.

Acceptance criteria: Runs a dummy HTTP server to serve pages for development testing.

Estimated completion date: Oct 17 2021 [Complete]

* 1. [Andi] Fetch HTML data from URL or local file path using Node-Fetch or FS
  2. [Andi] Process HTML data into a hierarchical structure using Cheerio

Priority: Tasks (b) and (c) are dependent on task (a) being completed, we want to finish this next since task (d) depends on having the HTML data in a tree structure which will be delivered in task (c).  
  
Acceptance criteria: Accepts a URL as a command line argument and fetches data from either a HTTP server or from a local file system. Given a 200 response, we should process the HTML content into a tree structure using cheerio, otherwise we can throw an error.   
  
Estimated completion date: October 18th 2021 [Complete]

* 1. [Yichen] Traverse hierarchical structure and extract only text content from tags, ignoring the hierarchical structure of data

Priority: This task is dependent on task (c), as this relies on the HTML tree structure created in task (c).

Acceptance criteria: Accepts a structure containing an HTML tree, then extracts the plaintext from the tree structure, and finally returns the plaintext.

Estimated completion date: Oct 19 2021 [Complete]

* 1. [Yichen] Write tests for parsing functions

Priority: This task will be made to test the correctness of task (d). This task is dependent on task (d).

Acceptance criteria: Thoroughly tests the extraction of plaintext for the function(s) in task (d).

Estimated completion date: Oct 20 2021 [Complete]

* 1. [Nicki] Dump extracted HTML text into the Blessed window

Priority: This task is dependent on task (b), (c), (d)

Acceptance criteria: Get the extracted web content and display it to the Blessed window as it is. Catch any server errors from task (b), (c) and display the errors on the Blessed window. Show the correct error message from the server without crashing the software.

Estimated completion date: Oct 23 2021 [Complete]

1. [In-progress] Terminal User Interface Class Hierarchy: this feature will result in the terminal user interface to be rewritten under-the-hood with a
   1. [Andi, Myles, Nicki, Yichen] Add functionality to backend processing class to format hierarchical structure based on their HTML tags, and display the formatted hierarchical structure in the frontend  
        
      Due to this task being so large and also critical to the rest of the tasks in milestone 3, we’ve included all members to ensure familiarity with the codebase which will help with subsequent tasks. Additionally all members will be solutioning this task together.

Acceptance criteria: Display the extracted web content in the right order of elements, and render bold/italics/underline tags with their corresponding terminal formatting. We must also verify text-clicking and scrolling are supported by our solution.

Estimated completion: Nov 16, 2021 [In Progress]

* 1. [Andi, Myles, Nicki, Yichen] Write tests for the updated backend functionality  
       
     Priority: Tests will be written for our utility functions from task (a) as they are being worked on. This task will be done in parallel with task (a).

Acceptance criteria: Display the extracted web content in the right order of elements, and render bold/italics/underline tags with their corresponding terminal formatting. Also, text-clicking and scrolling must be proven to be possible to implement.

Estimated completion: Nov 16, 2021 [In Progress]

* 1. [Andi] Add support in blessed client for clickable links  
       
     Priority: This task requires that our client display different HTML tags as separate box elements, as a result it will be scheduled for completion after task (a) is complete.  
       
     Acceptance criteria: Blessed client should display links, and links should be clickable with event handlers that redirect users to the link’s URL  
       
     Estimated completion date: Nov 17, 2021 [Not Started]
  2. [Nicki] Implement page scrolling for pages that are longer than the height of the terminal window

Priority: This task is dependent on task (c), we need to complete the frontend structure before implementing scrolling

Acceptance criteria: Smooth scrolling and displaying items in the correct order. Scroll up and down the screen using the mouse or keyboard.

Estimated completion date: Nov 17, 2021 [Not Started]

* 1. [Nicki] Handle errors when requesting from web servers (ex: 4XX, 5XX)

Acceptance criteria: When users enter a url which received an error from the server, the program will not crash. Display meaningful error messages or default messages received from the server to the user interface.

Estimated completion date: Nov 17, 2021 [Not Started]

# Success Criteria from Greg

As a team, we detailed tasks for our second and third milestones, and listed out what we believed to be important acceptance criteria. In order to align our industry partner’s vision of success with our own, we had several discussions with Greg where he ultimately approved the milestone plans we laid out. We have successfully completed ALL of the tasks in milestone 2 and validated their acceptance criterias with Greg. We are currently working on the tasks listed in milestone 3. We also discussed the tasks in our final milestone, where Greg provided some advice regarding our stretch goals.