

SPL-1 Project Report,2020

HTML to LaTeX Converter

SE 305: Software Project Lab 1

Submitted by :Mushfiquir Rahman

BSSE Roll No. : 1130

BSSE Session: 2018-19

Supervised By: Dr. B M Mainul Hossain

Designation: Associate Professor
Institute of Information Technology
University of Dhaka



Project: HTML to LaTeX converter
Author: Mushfiqur Rahman
Submitted: 26.08.21
Supervised By: Dr. B M Mainul Hossain
Associate Professor,
Institute of Information Technology
University of Dhaka


Supervisor's Approval:  26-08-2021

Table of contents

1.Introduction	4
1.1.Background Study	4
1.2 Challenges:	6
2.Objectives:	7
3.Scope:	7
4.Project Overview:	7
4.1 Get HTML from a webpage:	8
4.2 Create a tree by using HTML file:	9
4.3 Convert HTML into LaTeX :	10
5. User Manual:	11
6.Conclusion:	15
7.Appendix:	15
8.Reference :	16

1.Introduction

The HyperText Markup Language, or HTML is the standard markup language for documents designed to be displayed in a web browser. Web browser receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

LaTeX is a high-quality typesetting system; it includes features designed for the production of technical and scientific documentation. LaTeX is the de facto standard for the communication and publication of scientific documents.

The project is mainly a HTML to LaTeX converter. It will work with tree parsing for HTML files and convert tags and attributes into LaTeX equivalents. Project will get an html file from a given url and convert it to LaTeX format.

1.1.Background Study

To implement this project, some background study was necessary :

HTML:

I studied HTML and its structures prior to this project. I had to know about HTML tags, attributes and elements prior to this project. Also, I had to know about the similarity of HTML and LaTeX for this project.

LaTeX :

LaTeX is a document preparation system used for the communication and publication of scientific documents. Prior to this project, I had no knowledge about LaTeX. So, I had to study about LaTeX , LaTeX's structures ,its similarities with HTML and how to convert it into LaTeX.

DOM Tree Parsing:

Dom parsing differentiates the tags ,attributes and elements of HTML ,which was used in LaTeX conversion. I studied DOM tree parsing for this project .

Gumbo Parsing:

Gumbo is an implementation of the HTML5 parsing algorithm implemented as a pure C99 library with no outside dependencies. I studied about Gumbo parsing for my project and wanted to build my parser based around it.

Getting HTML from webpage:

To get html from a webpage in windows operating system,I have to study to do followings :

- Connect to server port 80
- Split the URL domain and its path
- Send HTTP GET Request
- Receive a reply from the server

1.2 Challenges:

Implementing a new software solution carries with it a number of challenges. The process can be overwhelming, confusing and lengthy. This is a new type of project for me and I didn't find any similar type of project in C languages. For implementing this project there are a lot of challenges that I have faced. Some of them are:

- Handling large c code for the first time
- Learning LaTeX for the first time
- Debugging a large number of codes was pretty challenging for me

- Getting HTML from a webpage was first for me
- There was a lot of trouble in memory allocation
- Tree parsing operation in C
- It was a challenge to decide on how to process HTML codes and how the approach should be to extract different features

2.Objectives:

HTML is the most important language for web development and LaTeX is also a very important language for scientific documentation . So, conversion between these two languages are very important.

In this project I made a simple HTML to LaTeX converter. Making it easy to use and efficient was my first priority.

3.Scope:

In this project, I am trying to convert HTML to LaTeX. For this ,I converted HTML tags and attributes to its equivalent LaTeX parts. Project also works in HTML tree parsing and shows about the parents and children of the tags.

Though there is a lot of work to be done. Still, there is some scope to develop the project. For example, there are some tags in HTML which haven't been converted to LaTeX yet in my project. Also, there are some memory constraints, So, I had to limit the number of tags that can be converted to LaTeX.

4.Project Overview:

I have divided my whole project into several parts:

Ø Get HTML from a webpage

Ø Create a tree by using HTML file

Ø Convert HTML into LaTeX

4.1 Get HTML from a webpage:

Here I used wininet library to get html from a webpage. First , I need to set up an internet session and then open a connection to a server and then send a get request and get a reply. In hConnection I put the url domain and in hData I put path,directories and webpage of the url. This will save HTML into a file with HTML extension and will also show how many bytes are received.

```
HINTERNET hInternet = InternetOpenA("Internet/1.0", INTERNET_OPEN_TYPE_PRECONFIG, NULL, NULL, 0);

HINTERNET hConnection = InternetConnectA( hInternet, pl.first.c_str(), 80, "", "", INTERNET_SERVICE_HTTP, 0, 0 ); //enter url here

HINTERNET hData = HttpOpenRequestA( hConnection, "GET", pl.second.c_str(), NULL, NULL, NULL, INTERNET_FLAG_KEEP_CONNECTION, 0 );

char buf[ 2048 ];

HttpSendRequestA( hData, NULL, 0, NULL, 0 );
string total;
DWORD bytesRead = 0;
DWORD totalBytesRead = 0;

while( InternetReadFile( hData, buf, 2000, &bytesRead ) && bytesRead != 0 )
{
    buf[ bytesRead ] = 0; // insert the null terminator.
    total+=total+buf;
    // printf( "%d bytes read\n", bytesRead );

    totalBytesRead += bytesRead ;
}
```

Figure 1: connect to server, send get request and reply

4.2 Create a tree by using HTML file:

A HTML tree is built based on tag, attribute and string element of tag. The root of this HTML tree is a tag. A Tag can have multiple children. But attribute and String will not have any children. This tree will be used to convert HTML to

LaTeX and also it will show the structure of the HTML webpage. For Example
:

```
<html>
  <head>
    <title>hello</title>
  </head>
  <body>
    <h1 align="center"><u>Mushfiqur Rahman</u></h1>
    <a href="http://www.google.com">This is a link</a>
  </body>
</html>
```

```
<html>
<head>
<title>
~hello
<body>
<h1>
!align="center"
<u>
~Mushfiqur Rahman
<a>
!href="http://www.google.com"
~This is a link
```

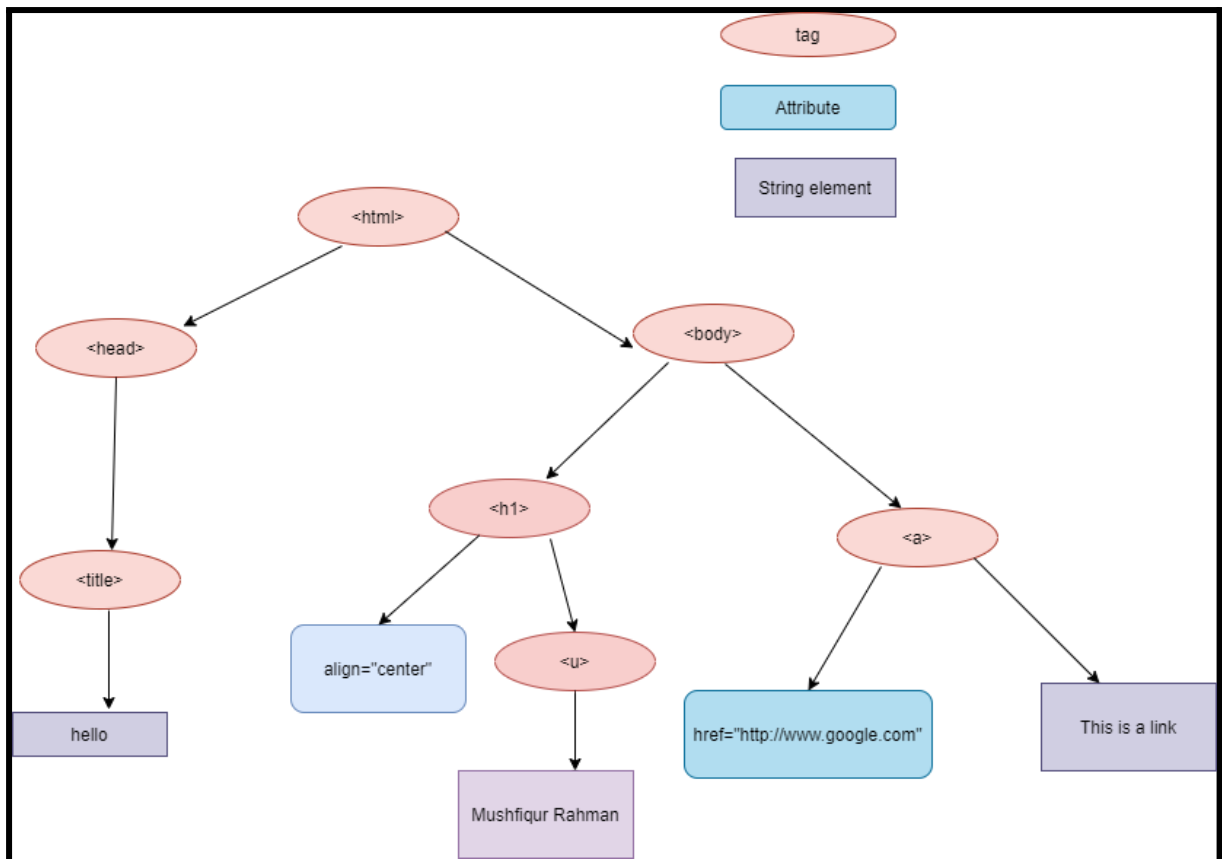



Figure 2:HTML parser tree

4.3 Convert HTML into LaTeX :

The project converts html tags to its' LaTeX equivalents.

for `` tag project converts it to `\textbf`

for `<i>` tag project converts it to `\itshape`

for `<u>` tag project converts it to `\underline`

for `<sup>` tag project converts it to `\textsuperscript`

for `<h1>` tag project converts it to `\section*`

for <h2> tag project converts it to \subsection*

for <h3> tag project converts it to \subsubsection*

for <h4> tag project converts it to \paragraph

for <h5> tag project converts it to \subparagraph

for <h6> tag project converts it to \subsection

for tag project converts it to \emph

for tag project converts it to \item

for <sub> project converts it to \subscript

for tag project converts it to \begin{itemize}

for tag project converts it to \begin{enumerate}

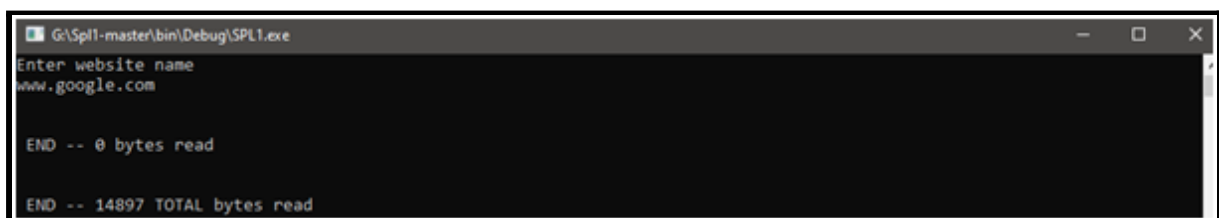
for tag project converts it to \includegraphics

for <a href> tag project converts it to \href

etc. and also converts HTML files' string element and attribute to their LaTeX equivalents. Such as, !href="link " will change to \href{link} . These are some of the conversations are made in this project

5. User Manual:

Open the project, build and run the program and enter Website name . You are recommended to use forward slash after inputting a website name.



```
(function(){google.jl={attn:false,blt:'none',chnk:0,dw:false,emtn:0,end:0,ine:false,lls:'default',pdt:0,rep:0,sif:true,snr:true,strt:0,ubm:false,uwp:true;}});(function(){var pmc='{\\x22d\\x22:{},\\x22sb_he\\x22:{\\x22agen\\x22:true,\\x22cgen\\x22:true,\\x22client\\x22:\\x22heirloom-hp\\x22,\\x22dh\\x22:true,\\x22dhqt\\x22:true,\\x22ds\\x22:\\x22\\x22,\\x22ffgl\\x22:\\x22en\\x22,\\x22fl\\x22:true,\\x22host\\x22:\\x22google.com\\x22,\\x22isbh\\x22:28,\\x22jsonp\\x22:true,\\x22msgs\\x22:{\\x22cibl\\x22:\\x22&#2488; &#2494; &#2480; &#2509; &#2458; &#2488; &#2494; &#2475; &#2453; &#2480; &#2497; &#2472;\\x22,\\x22dym\\x22:\\x22&#2438; &#2474; &#2472; &#2495; &#2453; &#2495; &#2476; &#2507; &#2461; &#2494; &#2468; &#2503; &#2458; &#2503; &#2527; &#2503; &#2459; &#2503; &#2472;\\x22,\\x22lcky\\x22:\\x22&#2477; &#2494; &#2455; &#2509; &#2479; &#2476; &#2494; &#2472; &#2437; &#2472; &#2497; &#2477; &#2476; &#2453; &#2480; &#2459; &#2495;\\x22,\\x22lml\\x22:\\x22&#2438; &#2480; &#2451; &#2460; &#2494; &#2472; &#2497; &#2472;\\x22,\\x22oskt\\x22:\\x22&#2439; &#2472; &#2474; &#2497; &#2463; &#2463; &#2497; &#2482;\\x22,\\x22psrc\\x22:\\x22&#2437; &#2472; &#2497; &#2488; &#2472; &#2509; &#2471; &#2494; &#2472; &#2472; &#2494; &#2472; &#2463; &#2495; &#2438; &#2474; &#2472; &#2494; &#2480; \\u003Ca href\\x3d\\x22/history\\x22\\u003E&#2451; &#2527; &#2503; &#2476; &#2439; &#2468; &#2495; &#2489; &#2494; &#2488; \\u003C/a\\u003E &#2469; &#2503; &#2453; &#2503; &#2488; &#2480; &#2494; &#2472; &#2507; &#2459; &#2495; &#2482;\\x22,\\x22psrl\\x22:\\x22&#2488; &#2480; &#2494; &#2472;\\x22,\\x22sbit\\x22:\\x22&#2459; &#2476; &#2495; &#2437; &#2472; &#2497; &#2488; &#2494; &#2480; &#2503; &#2488; &#2472; &#2509; &#2471; &#2494; &#2472; &#2453; &#2480; &#2497; &#2472;\\x22,\\x22srch\\x22:\\x22Google Search\\x22},\\x22ovr\\x22:{},\\x22pq\\x22:\\x22\\x22,\\x22refpd\\x22:true,\\x22rfs\\x22:[],\\x22sbas\\x22:\\x220 3px 8px 0 rgba(0,0,0,0.2),0 0 0 1px rgba(0,0,0,0.08)\\x22,\\x22sbpl\\x22:16,\\x22sbpr\\x22:16,\\x22scd\\x22:10,\\x22stok\\x22:\\x223YxalXP0zGIVHD_t_p8qlNaor1A\\x22,\\x22uhde\\x22:false}}';google.pmc=JSON.parse(pmc);})();</script> </body></html>
Press any key to continue . . .
```

Figure 3:Enter a website name and get html file

This will give an HTML file of the website and press any key to continue. After pressing any key, the console will show HTML structure and also total number of tag counts and some statistics.

```
~ne110
<i>
~Lorem ipsum
<b>
~and this is bold text
<em>
~This text is emphasized
<u>
~This text will be underlined.
<sup>
~superscripted
<sub>
~subscripted
<h2>
~Heading 2
<h3>
~Heading 3
<h4>
~Heading 4
<h5>
~Heading 5
<h6>
~Heading 6
total tag count 22
total link 1
```

Figure 4:Shows Html files' structure

Press 1 to see a tag's parents and it will show Enter tag name. After putting the child's tag name ,it will show its parent tag.

```

1.tag's parent
2.tag's children
6.exit
1
Enter Tag Name : <a>
<a>'s parent is<body>

```

Figure 5:showing parent of a tag

Press 2 to see a tag's child and it will show Enter tag name. After putting the parent's tag name ,it will show its children tag.

```

1.tag's parent
2.tag's children
6.exit
2
Enter Tag Name : <p>
<p>'s children are<b>

```

Figure 6:showing Child of a tag

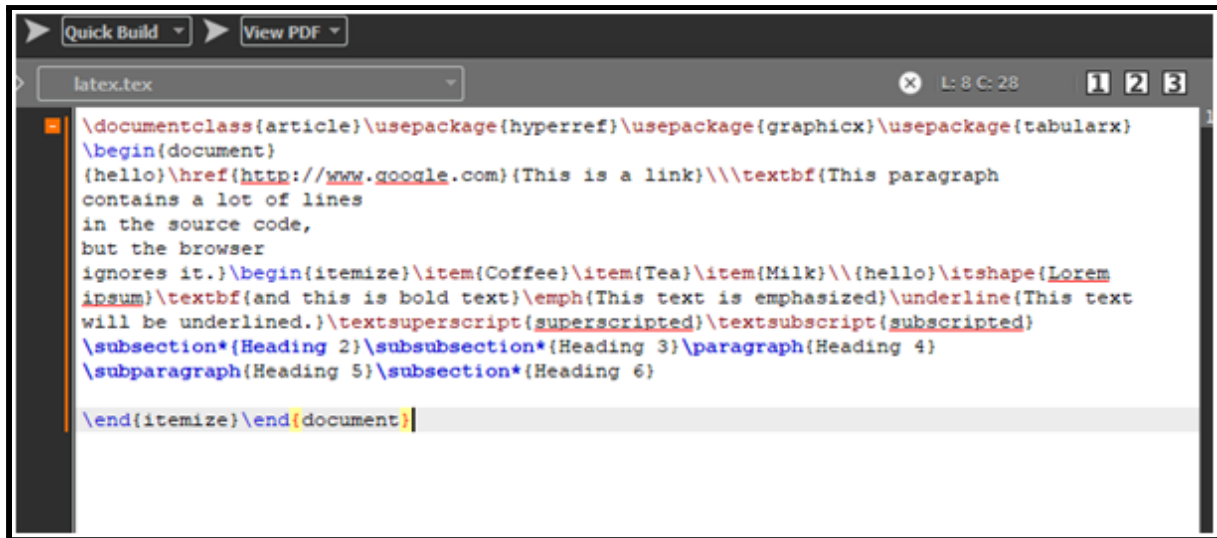
Press 3 to show tag's siblings who are children of same parent tag

```

1.tag's parent
2.tag's children
3.tag's siblings
6.exit
3
Enter Tag Name : <p>
Siblings are <a>      Siblings are <p>      Siblings are <ul>      Siblings are <p>      Siblings are <i>
Siblings are <b>      Siblings are <em>      Siblings are <u>      Siblings are <sup>      Siblings are <sub>

```

Figure 7:showing siblings of a tag



```

\documentclass{article}\usepackage{hyperref}\usepackage{graphicx}\usepackage{tabularx}
\begin{document}
(hello)\href{http://www.google.com}{This is a link}\\\textbf{This paragraph
contains a lot of lines
in the source code,
but the browser
ignores it.}\begin{itemize}\item{Coffee}\item{Tea}\item{Milk}\\\{hello}\itshape{Lorem
ipsum}\textbf{and this is bold text}\emph{This text is emphasized}\underline{This text
will be underlined.}\textsuperscript{superscripted}\textsubscript{subscripted}
\subsection*{Heading 2}\subsubsection*{Heading 3}\paragraph{Heading 4}
\subparagraph{Heading 5}\subsection*{Heading 6}

\end{itemize}\end{document}

```

figure 8:LaTeX file of the webpage

Do compile and quick build and you will see a LaTeX version of the webpage

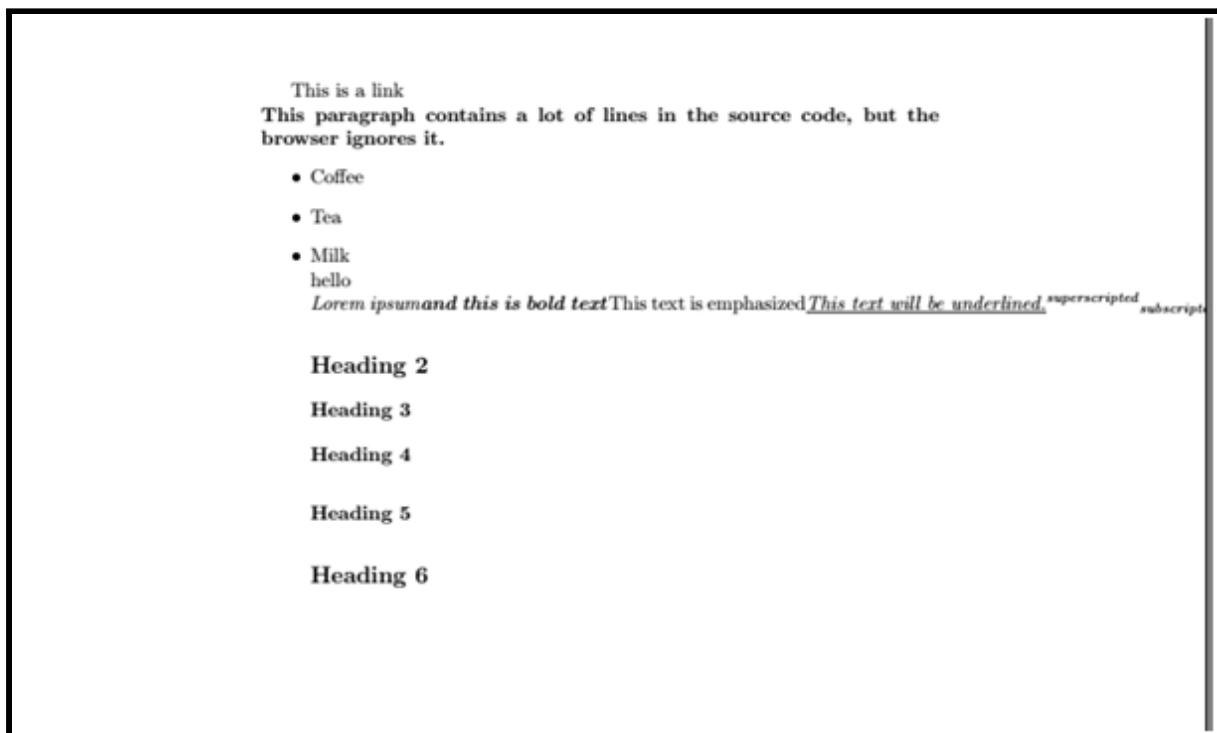


figure 9:LaTeX version of the webpage

6.Conclusion:

The project helped me to learn about html parsing and LaTeX. Working with parsing was quite challenging and I had to do a lot of background study in parsing and LaTeX. First time,I maintained a large number of codes and that was quite challenging. Besides,I have faced a lot of problems and tried to solve all of them. So,I learned a lot about C programming. Also I had to learn about socket programming and http get requests. For the first time ,I used wininet Library . I am happy that I have taken the project to a state of progress that I had hoped at the beginning. Overall ,it was a great experience and I really look forward to using this experience in future projects and programming works .

7.Appendix:

Converting from HTML to LaTeX is a long process, so I have done only for important and common tags. I also made the project for the whole HTML file but due to memory constraints I have to do the conversation for a limited number of tags. I will look forward to improving on these aspects.

8.Reference :

Gumbo Parser:

<https://github.com/google/gumbo-parser?fbclid=IwAR2pUOLMWEUIAoP3BaCeRH3n0riwwesBbLQacODbCyVA1Kxj1pNN44Y7q6s>

LaTeX Learning:

https://www.overleaf.com/learn/latex/Learn_LaTeX_in_30_minutes

DOM Parsing:

<https://www.w3.org/TR/DOM-Parsing/>

HTML tutorial:

<https://www.w3schools.com/html/>

<https://www.tutorialspoint.com/html/index.htm>