Kiosk User Guide

Team 1086 Blue Cheese Robotics

Sophia Larson



Contents

[Kiosk Components and Organization 3](#_Toc6826254)

[Page Structure 3](#_Toc6826255)

[File Structure 4](#_Toc6826256)

[Media Resources 7](#_Toc6826257)

[Hardware Components 9](#_Toc6826258)

[How to edit 10](#_Toc6826259)

[Presentations 10](#_Toc6826260)

[Images 10](#_Toc6826261)

[Words 10](#_Toc6826262)

[Steps for Use 11](#_Toc6826263)

[Common Problems 12](#_Toc6826264)

[File Not Found 12](#_Toc6826265)

[Image Not Displaying 12](#_Toc6826266)

[Something is in front of another element 12](#_Toc6826267)

[Content not displaying 12](#_Toc6826268)

[Image is fuzzy 12](#_Toc6826269)

[Page slow opening 12](#_Toc6826270)

[Other Issues 12](#_Toc6826271)

[Page-By- Page Explanation 13](#_Toc6826272)

[Start 13](#_Toc6826273)

[Overview 13](#_Toc6826274)

[Layout 13](#_Toc6826275)

[Code 16](#_Toc6826276)

[FIRST 18](#_Toc6826277)

[Overview 18](#_Toc6826278)

[Layout 18](#_Toc6826279)

[Code 22](#_Toc6826280)

[Impact 28](#_Toc6826281)

[Overview 28](#_Toc6826282)

[Layout 28](#_Toc6826283)

[Code 32](#_Toc6826284)

[Levels 40](#_Toc6826285)

[Overview 40](#_Toc6826286)

[Layout 40](#_Toc6826287)

[Code 46](#_Toc6826288)

[Team 53](#_Toc6826289)

[Overview 53](#_Toc6826290)

[Layout 53](#_Toc6826291)

[Code 57](#_Toc6826292)

[About 62](#_Toc6826293)

[Overview 62](#_Toc6826294)

[Layout 62](#_Toc6826295)

[Code 66](#_Toc6826296)

[Sub teams 74](#_Toc6826297)

[Overview 74](#_Toc6826298)

[Layout 74](#_Toc6826299)

[Code 81](#_Toc6826300)

[Accomplishments 89](#_Toc6826301)

[Overview 89](#_Toc6826302)

[Layout 89](#_Toc6826303)

[Code 93](#_Toc6826304)

[Outreach 101](#_Toc6826305)

[Overview 101](#_Toc6826306)

[Layout 101](#_Toc6826307)

[Code 105](#_Toc6826308)

[Competitions 122](#_Toc6826309)

[Overview 122](#_Toc6826310)

[Layout 122](#_Toc6826311)

[Code 125](#_Toc6826312)

[Glossary 128](#_Toc6826313)

# Kiosk Components and Organization

## Page Structure

## File Structure

KIOSK

* bgtile.jpg
* Resources 📁[[1]](#footnote-1)
  + About.pptx
  + Awards.pptx
  + Outreach.pptx
* Start.html
* Start 📁
  + cheese\_world.gif
* FIRST.html
* FIRST 📁
  + FIRST.png
* Impact.html
* IMPACT 📁
  + poster1a
  + poster1b
  + poster2a
  + poster2b
  + poster3a
  + poster3b
  + poster4a
  + poster4b
* Levels.html
* Levels 📁
  + FRCVECTOR.svg
  + FLLVECTOR.svg
  + FLLJRVECTOR.svg
  + FTCVECTOR.svg
* TEAM.html
* TEAM 📁
  + TEAM.svg
  + SliceLife.gif
* About.html
* about\_slides 📁
  + Slide1.png
  + Slide2.png
  + Slide3.png
  + Slide4.png
  + Slide5.png
  + Slide6.png
* Subteams.html
* Subteams 📁
  + CAD.png
  + Fabrications.JPG
  + build.png
  + Programming.JPG
  + media.JPG
  + Chairmans.JPG
* Acccomplishments.html
* awards\_slides 📁
  + Slide1.png
  + Slide2.png
  + Slide3.png
  + Slide4.png
  + Slide5.png
  + Slide6.png
  + Slide7.png
  + Slide8.png
* Outreach.html
* Outreach 📁
* outreach\_slides 📁
  + Slide2.PNG
  + Slide3.PNG
  + Slide4.PNG
  + Slide5.PNG
  + Slide6.PNG
  + Slide7.PNG
  + Slide8.PNG
  + Slide9.PNG
  + Slide10.PNG
  + Slide11.PNG
  + Slide12.PNG
  + Slide13.PNG
  + Slide14.PNG
  + Slide15.PNG
  + Slide16.PNG
  + Slide17.PNG
  + Slide18.PNG
  + Slide19.PNG
  + Slide20.PNG
  + Slide21.PNG
  + Slide22.PNG
  + Slide23.PNG
  + Slide24.PNG
  + Slide25.PNG
  + Slide26.PNG
  + Slide27.PNG
  + Slide28.PNG
  + Slide29.PNG
  + Slide30.PNG
  + Slide31.PNG
  + Slide32.PNG
  + Slide33.PNG
  + Slide34.PNG
  + Slide35.PNG
  + Slide36.PNG
  + Slide37.PNG
  + Slide38.PNG
  + Slide39.PNG
  + Slide40.PNG
  + Slide41.PNG
  + Slide42.PNG
  + Slide43.PNG
  + Slide44.PNG
  + Slide45.PNG
  + Slide46.PNG
  + Slide47.PNG
  + Slide48.PNG
  + Slide49.PNG
  + Slide50.PNG
  + Slide51.PNG
  + Slide52.PNG
  + Slide53.PNG
  + Slide54.PNG
  + Slide55.PNG
  + Slide56.PNG
  + Slide57.PNG
  + Slide58.PNG
  + Slide59.PNG
  + Slide60.PNG
* Competitions.html
* Competitions 📁
  + GameManual.pdf

## Media Resources

* Pictures
  + FIRST.png
  + poster1a.jpg
  + poster1b.jpg
  + poster2a.jpg
  + poster2b.jpg
  + poster3a.jpg
  + poster3b.jpg
  + poster4a.jpg
  + poster4b.jpg
  + FRCVECTOR.svg
  + FLLVECTOR.svg
  + FLLJRVECTOR.svg
  + FTCVECTOR.svg
  + TEAM.svg
  + CAD.png
  + Fabrications.JPG
  + build.png
  + Programming.JPG
  + media.JPG
  + Chairmans.JPG
  + In About 📁
    - Slide1.png
    - Slide2.png
    - Slide3.png
    - Slide4.png
    - Slide5.png
    - Slide6.png
  + In awards 📁
    - Slide1.png
    - Slide2.png
    - Slide3.png
    - Slide4.png
    - Slide5.png
    - Slide6.png
    - Slide7.png
    - Slide8.png
  + In Outreach 📁
    - Slide2.PNG
  + Slide3.PNG
  + Slide4.PNG
  + Slide5.PNG
  + Slide6.PNG
  + Slide7.PNG
  + Slide8.PNG
  + Slide9.PNG
  + Slide10.PNG
  + Slide11.PNG
  + Slide12.PNG
  + Slide13.PNG
  + Slide14.PNG
  + Slide15.PNG
  + Slide16.PNG
  + Slide17.PNG
  + Slide18.PNG
  + Slide19.PNG
  + Slide20.PNG
  + Slide21.PNG
  + Slide22.PNG
  + Slide23.PNG
  + Slide24.PNG
  + Slide25.PNG
  + Slide26.PNG
  + Slide27.PNG
  + Slide28.PNG
  + Slide29.PNG
  + Slide30.PNG
  + Slide31.PNG
  + Slide32.PNG
  + Slide33.PNG
  + Slide34.PNG
  + Slide35.PNG
  + Slide36.PNG
  + Slide37.PNG
  + Slide38.PNG
  + Slide39.PNG
  + Slide40.PNG
  + Slide41.PNG
  + Slide42.PNG
  + Slide43.PNG
  + Slide44.PNG
  + Slide45.PNG
  + Slide46.PNG
  + Slide47.PNG
  + Slide48.PNG
  + Slide49.PNG
  + Slide50.PNG
  + Slide51.PNG
  + Slide52.PNG
  + Slide53.PNG
  + Slide54.PNG
  + Slide55.PNG
  + Slide56.PNG
  + Slide57.PNG
  + Slide58.PNG
  + Slide59.PNG
  + Slide60.PNG
* GIF
  + cheese\_world.gif
  + SliceLife.gif
* PowerPoints
  + About.pptx
  + Awards.pptx
  + Outreach.pptx
* PDF
  + GameManual.pdf

## Hardware Components

* Asus two-in-one laptop
* Asus Charger
* Extension Cord
* Lockable Strap
* Stand

# How to edit

## Presentations

* Find files in the resource section that correspond to the presentation you want to change. Make the changes you want. When you are done got to Export-> Change File Type->PNG. Either select to export just the one slide or the whole PowerPoint. Put these new files in replacement of the files within the code making sure they have the same file name.

## Images

* If it is not a PowerPoint, find the src=”” of the img you want to change. Change the file in the folder to the new content you want, but make sure to keep the naming the same.

## Words

* Words displayed are shown between tags for a text element using the commented code for assistance. often in <a>, <p>, <h1>, or <h2>. Change the words in order to change the words in the kiosk

# Steps for Use

1. Open the File Start.html in Google Chrome
2. Press F11
3. Flip Around the keyboard and the laptop will enter tablet mode
4. Place the tablet on the stand and place the lockable strap in between the folded laptop and around the stand.
   1. The lockable straps password is the last three digits of the year we won worlds.
5. If the kiosk is going to be used for more than five hours, plug in the power cord into an outlet using the power strip if necessary.
6. Open up once all pages (in order for a faster load when in use)
7. Keep an eye on kiosk when in use
8. When kiosk needs to be not in use slide the cover over both the laptop and stand
9. To put away detach the laptop and press F11 once more and put away all parts of the kiosk

­­

# Common Problems

## File Not Found

* Check that the file name matches what it is supposed to say. (See File Structure) Make sure the file exists. Check within the code that the marked button links to what it is supposed to link to.

## Image Not Displaying

* Check that the file name matches what it is supposed to say. (See File Structure) Make sure the file exists. Check within the code that the marked img href=”” to what it is supposed to link to.

## Something is in front of another element

* Check that the z-index. The element with the greatest z-index appears first. This will only work if it has a position: \_\_ declared.

## Content not displaying

* Open the page in chrome and press CTRl + I or right click and click inspect. Check that comments displayed in green don’t overlap. Check that is has an opening and closing tag. To find missing things within elements check that the color styling matches. Things that are the same will have the same color.

## Image is fuzzy

* Either attempt to make it an svg or make the img display a smaller size. You can only make it an SVG if it is a digitally created img.

## Page slow opening

* The computer will access files quicker if you open everything once to start. Go though and open everything to make sure it has all loaded. Make sure that no other programs are running

## Other Issues

* Press Ctrl+ I with the page open and you will see displayed many information to check matches what the kiosk should be. Below are sites to got to for help
  + <https://www.w3schools.com/>
  + <https://stackoverflow.com/>
  + <http://www.simplehtmlguide.com/cheatsheet.php>
  + <http://www.simplehtmlguide.com/cheatsheet.php>

# Page-By- Page Explanation

## Start

### Overview

Content: Has a basic opening gif on loop and buttons for the three next subpages.

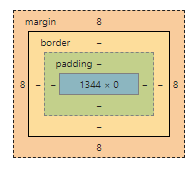
Purpose: In order to be a page to start from.

Audience: Meant for all users.

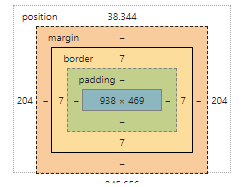
### Layout



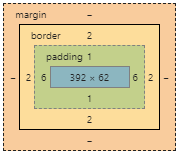
1. body



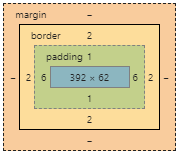
1. Main\_Image



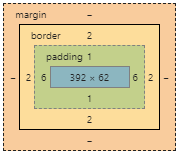
1. button1



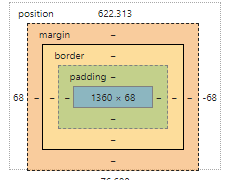
1. button2



1. button3



1. buttons



### Code

<!DOCTYPE html>

<html lang="en"> ***<!--set up that this is an html file and the language is english-->***

<head>

<meta charset="UTF-8"> ***<!--defines the type of charcters being used-->***

<style>

/\*sets settings and background\*/

\* { ***/\*all sellector\*/***

box-sizing: border-box; ***/\*The CSS box-sizing property allows us to include the padding and border in an element’s total width and height. The border box variable makes it so element padding and border are included in the width and height\*/***

}

body { ***/\*effectivly all selector\*/***

background-image: url("bgtile.jpg"); ***/\*sets the background image\*/***

}

***/\* CSS for the Central Image \*/***

.Main\_Image { /\*selector for image\*/

position:absolute; ***/\*The position property specifies the type of positioning method used for an element. The absolute variable means The element is positioned relative to anything it is containted within. In this case the body. \*/***

border: 7px solid #fee622;

width: 70%; ***/\*sets the width to 70%\*/***

top:5%; ***/\*moves it 5% from the top\*/***

left:15%; ***/\*moves it 15% from the left\*/***

}

***/\*CSS for the buttons\*/***

button { ***/\*selects all buttons\*/***

background-color: black;

color: white;

cursor: pointer; ***/\*sets the cursor to the pointer hand while cursor over it\*/***

font-size: 40px;

font-family: "Lucida Sans Unicode"; ***/\*sets the font\*/***

}

.buttons {

position:fixed; ***/\*The position property specifies the type of positioning method used for an element. The fixed variable means that the element is positioned relative to the browser window\*/***

left:5%; ***/\*moves it 5% from the left\*/***

bottom:10%; ***/\*moves it 10% from the bottom\*/***

}

</style>

</head>

<body>

<img class= "Main\_Image" alt="" src="Start/cheese\_world.gif"> ***<!--inserts the image it has a class that is used to be called upon in css as.Main\_Image and get the image from src=""-->***

<div class = "buttons" style="width:100%"> ***<!--Defines the container for the buttons-->***

<button class="button1" style="width:30%;" onclick="window.location.href = 'FIRST.html';">FIRST</button***><!-- for both of theses it defines what the buttons are called in css the width src="" is where the button takes them and the words in between the opener and closer are the words displayed on the button-->***

<button class="button2" style="width:30%;" onclick="window.location.href = 'Team.html';">OUR TEAM</button>

<button class="button3" style="width:30%;" onclick="window.location.href = 'Competition.html';">COMPETITION</button>

</div>

</body>

</html>

## FIRST

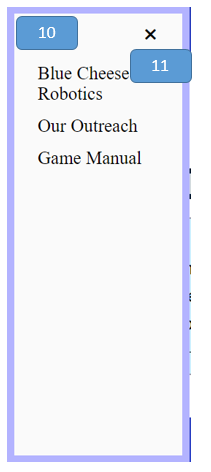
### Overview

Content: Contains the FIRST mission statement and links to the sub pages.

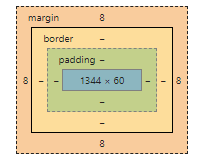
Purpose: Is a first introduction to first before the user delves deeper

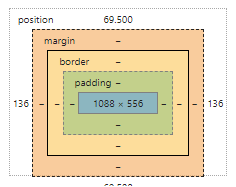
Audience: Users with little to none prior experience with FIRST.

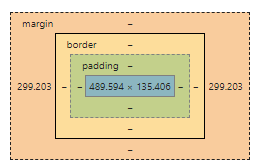
### Layout



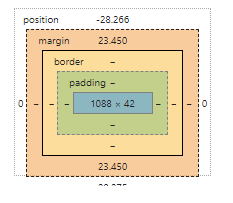
1. body



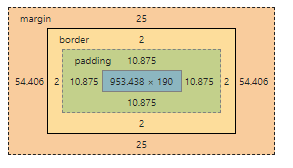
1. Main\_Content
   1. 
2. Main\_Image



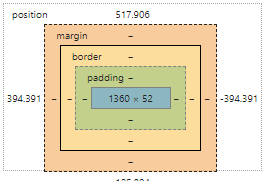
1. Content1



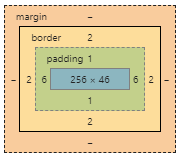
1. Content2



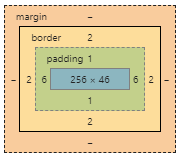
1. buttons



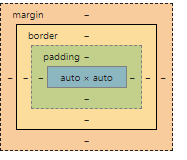
1. button1



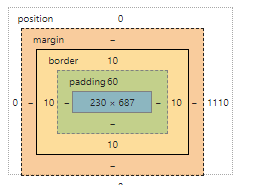
1. button2



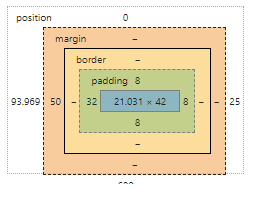
1. menu



1. Menu



1. close



### Code

<!DOCTYPE html>

<html lang="en"> ***<!--set up that this is an html file and the language is English-->***

<head>

<meta charset="UTF-8"> ***<!--defines the type of characters being used-->***

<style>

***/\*sets settings and background\*/***

\* { ***/\*all selector\*/***

box-sizing: border-box; /***\*The CSS box-sizing property allows us to include the padding and border in an element’s total width and height. The border box variable makes it so element padding and border are included in the width and height\*/***

}

body { ***/\*effectively all selector\*/***

background-image: url("bgtile.jpg"); ***/\*sets the background image\*/***

}

***/\*CSS for the main content area\*/***

.Main\_Content { ***/\*selector for main content area\*/***

position: absolute; ***/\*The position property specifies the type of positioning method used for an element. The absolute variable means The element is positioned relative to anything it is contained within. In this case the body. \*/***

width:80%; ***/\*sets the width to 80%\*/***

height:80%; ***/\*sets the height to 80%\*/***

right:10%; /***\*puts it 10% from the right\*/***

top:10%; ***/\*sets it 10% from the top\*/***

background-color: white;

}

***/\* CSS for the Central Image \*/***

.Main\_Image { /***\*selector for image\*/***

display: block; ***/\*The display property specifies the display behavior (the type of rendering box) of an element. The block variable means it displays an element as a block element (like <p>). It starts on a new line, and takes up the whole width\*/***

margin-left: auto***;/\*effectively this in combination with the display setting centers an element\*/***

margin-right: auto;

width: 45%;/***\*sets width to 45%\*/***

top:20%; /***\*moves it 20% from the top\*/***

}

***/\*CSS for the buttons\*/***

.Main\_Content .buttons***{ /\*selector that positions buttons in terms of being withing the main content\*/***

position:fixed***; /\*The position property specifies the type of positioning method used for an element. The fixed variable means that the element is positioned relative to the browser window\*/***

left:29%; ***/\*moves it 29% from the left\*/***

bottom:18%; /***\*moves it 18% from the bottom\*/***

}

button { ***/\*selects all buttons\*/***

background-color: black;

color: white;

cursor: pointer; ***/\*sets the cursor to the pointer hand while cursor over it\*/***

font-size: 30px;

font-family: "Lucida Sans Unicode"; ***/\*sets the font\*/***

box-sizing: border-box; ***/\*The CSS box-sizing property allows us to include the padding and border in an element’s total width and height. The border box variable makes it so element padding and border are included in the width and height\*/***

}

button1{ ***/\*selects only first button but it's effect positions the whole\*/***

left:30%; ***/\*moves it to the left 30%\*/***

}

***/\*CSS for the side menu bar\*/***

***/\* Set up the side bar menu itself. Javascript will make it so that it will appear by chaning the width from zero \*/***

.Menu { /\****selects the overall menu section\*/***

height: 100%; ***/\*sets the height to 100%\*/***

width: 0; ***/\*changed in javascript when you click to open the menu\*/***

position: fixed; /***\*The position property specifies the type of positioning method used for an element. The fixed variable means that the element is positioned relative to the browser window\*/***

z-index: 4; ***/\*The z-index property specifies the stack order of an element. An element with greater stack order is always in front of an element with a lower stack order.\*/***

top: 0; ***/\*puts it at the top\*/***

left: 0; ***/\*puts it to the left\*/***

background-color: #fafafa;

overflow-x: hidden; ***/\*The overflow-x property specifies whether to clip the content, add a scroll bar, or display overflow content of a block-level element, when it overflows at the left and right edges. The hidden variable makes it so the content is clipped - and no scrolling mechanism is provided\*/***

transition: 0.2s; ***/\*Specifies how many seconds or milliseconds the transition effect takes to complete. In this case the transition from 0 width to full side bar width\*/***

padding-top: 60px; : ***/\*Padding creates extra space within an element on the top\*/***

}

***/\*Design for the words within the side bar menu\*/***

.Menu a { ***/\*selects the words within the side bar menu\*/***

padding: 8px 8px 8px 32px; ***/\* sets extra space within the element in the order of top, right, bottom, then left\*/***

text-decoration: none***; /\*takes away the standard underline in a link\*/***

font-size: 25px;

color: black;

display: block; /***\*The display property specifies the display behavior (the type of rendering box) of an element. The block variable means it displays an element as a block element (like <p>). It starts on a new line, and takes up the whole width\*/***

transition: 0.1s; ***/\*Specifies how many seconds or milliseconds the transition effect takes to complete. In this case the transition form 0 width to full side bar width\*/***

}

***/\*Positions and designs the menu close button\*/***

.Menu .close { ***/\*selects the close icon within the menu sidebar\*/***

position: absolute; ***/\*The position property specifies the type of positioning method used for an element. The absolute variable means The element is positioned relative to anything it is containted within. In this case the sidebar menu. \*/***

top: 0; ***/\*puts it at the top\*/***

right: 25px; ***/\*puts it that amount from the right\*/***

font-size: 36px;

font-family: Arial***;/\*sets the font\*/***

margin-left: 50px; ***/\*margin creates extra space around an element in this case to the left\*/***

}

***/\*CSS for the written content\*/***

***/\*Positions and designs the first section of content under the image\*/***

.content1 { ***/\*selects the first content section under the image\*/***

text-align: center; ***/\*The text-align property specifies the horizontal alignment of text in an element. In this case it is to the center\*/***

position:relative; ***/\*The position property specifies the type of positioning method used for an element. The variable realtive means the element is positioned relative to its normal position, so "left:20px" adds 20 pixels to the element's LEFT position\*/***

top:-10%; ***/\*makes it higher than it would usually be\*/***

font-family: Impact, Charcoal, sans-serif; ***/\*sets the font\*/***

font-size:35px;

text-shadow***:/\* this creats a shadow under the text. This creates a shadow in each side in the order of top, right, botton, then left. It is defined by the horizantal shadow amoumt, the vertical shadow amount, the blur(not important here), and the color of the shadow\*/***

-1px -1px 0 #01264b,

1px -1px 0 #01264b,

-1px 1px 0 #01264b,

1px 1px 0 #01264b;

}

***/\*Positions and designs the second section of content under the image\*/***

.content2 { ***/\*selects the second content section under the image\*/***

text-indent: 50px***;/\*The text-indent property specifies the indentation of the first line in a text-block.\*/***

text-align: justify; /\****The text-align property specifies the horizontal alignment of text in an element. The jusify variable means it stretches the lines so that each line has equal width (like in newspapers and magazines)\*/***

border: 2px solid #9d9daf; ***/\*this creates a border around the element it is define by width, what type, then color\*/***

border-radius: 8px; ***/\*this rounds the border around an element\*/***

padding: 1%; ***/\*Padding creates extra space within an element\*/***

width: 90%; ***/makes the width 90%\*/***

display: block; ***/\*The display property specifies the display behavior (the type of rendering box) of an element. The block variable means it displays an element as a block element (like <p>). It starts on a new line, and takes up the whole width\*/***

margin-left: auto; ***/\*effectly this in combination with the dispaly setting centers an element\*/***

margin-right: auto;

font-size:25px;

font-family: "Lucida Sans Unicode", "Lucida Grande", sans-serif***;/\*sets the font\*/***

}

</style>

</head>

<body>

<div id="Menu" class="Menu"> <!--sets up the container for the side bar menu-->

<a href="javascript:void(0)" class="close" onclick="closeMenu()">&times;</a> ***<!-created the close icon that on click actiacted the closeMenu() function. The symbol is created using a char code-->***

<a href="Team.html">Blue Cheese Robotics</a> ***<!--makes the words take the user to the defind href="" location when clicked-->***

<a href="Outreach.html">Our Outreach</a> ***<!--makes the words take the user to the defind href="" location when clicked-->***

<a href="Competition.html">Game Manual</a> ***<!--makes the words take the user to the defind href="" location when clicked-->***

</div>

<span style="font-size:45px;cursor:pointer;color:white;font-family: Arial" onclick="openMenu()">&#9776; Menu</span> ***<!--styles the open menu icon including making the cursor a pointer while above the element, sets the font, and calls the function openMenu() to open the menu when clicked. The wierd charcers infornt of menu makes the three bars symbol.-->***

<script> ***<!--defines when javascript starts-->***

function openMenu() { ***/\*function called when the user clicks on the span with the menu in the corner\*/***

document.getElementById("Menu").style.width = "250px"; ***<!--changes the width to make the side bar slide out assisted with the transition in css-->***

document.getElementById("Menu").style.border = "10px solid #b3b3ff"; ***<!--makes the border around the element come with the sidebar-->***

}

function closeMenu() {

document.getElementById("Menu").style.width = "0"; ***<!--changes the width to make the side bar slide in assisted with the transition in css-->***

document.getElementById("Menu").style.border = "0px"; ***<!--makes it so the border goes away when the side bar goes away-->***

}

</script> ***<!--defines when javascript ends-->***

<div class= "Main\_Content"> ***<!-- container for the actual content of the page-->***

<img id ="title" class= "Main\_Image" src="FIRST/FIRST.png" alt=""> ***<!--inserts the image it has a class that is used to be called upon in css as.Main\_Image and get the image from src=""-->***

<div class="content"> ***<!--sets the container for the written conntent-->***

<h1 class="content1" style="background-color:#011a32;font-family: Impact, Charcoal, sans-serif;color: #d1d9e0;">For Inspiration and Recognition of Science and Technology</h1>

***<!-- has the first content uder thimage also set the background color, font, and color, the thing with in the opening and closeing is the words shown-->***

<p class="content2" style="background-color:#cceeff;color:black;">The mission of FIRST&copy; is to inspire young people to be science and technology leaders and innovators, by engaging them in exciting mentor-based programs that build science, engineering, and technology skills, that inspire innovation, and that foster well-rounded life capabilities including self-confidence, communication, and leadership.</p>

***<!-- has the first content uder thimage also set the background color, font, and color, the thing with in the opening and closeing is the words shown-->***

</div>

<div class = "buttons" style="width:100%"> ***<!--Defines the container for the buttons-->***

<button class="button1" style="width:20%;" onclick="window.location.href = 'Impact.html';">Impact</button> ***<!-- for both of theses it defines what the buttons are called in css the width src="" is where the button takes them and the words in between the opener and closer are the words displayed on the button-->***

<button class="button2" style="width:20%;" onclick="window.location.href = 'Levels.html';">Levels</button>

</div>

</div>

</body>

</html>

## Impact

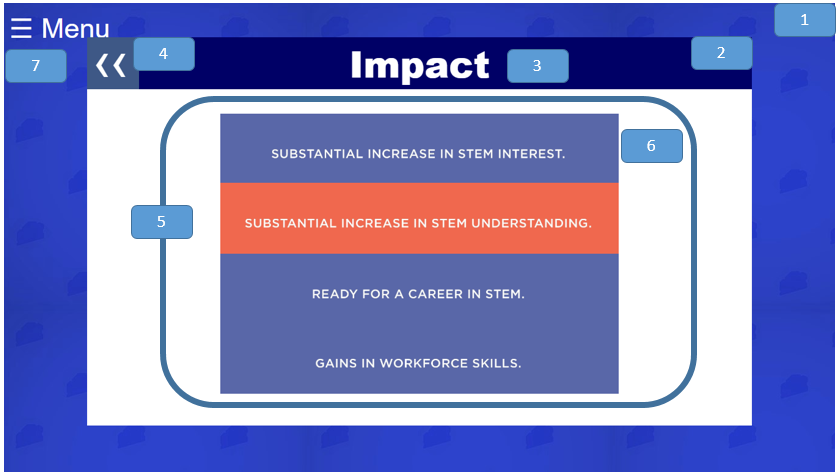
### Overview

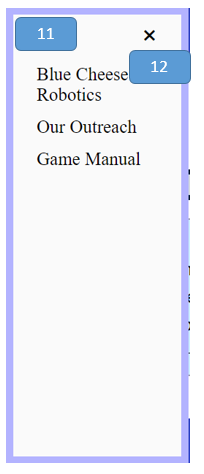
Content: Interactive infographic of the impact of FIRST

Purpose: To convey the importance of FIRST and its impact on participants.

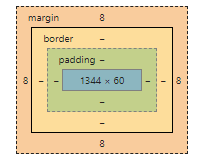
Audience: Users with little to none prior experience with FIRST.

### Layout

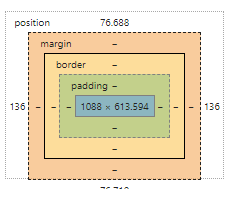




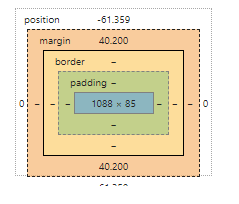
1. body



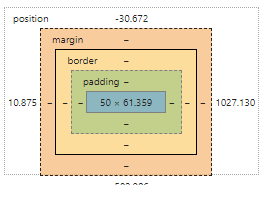
1. Main\_Content

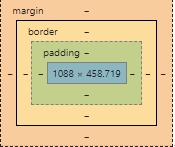


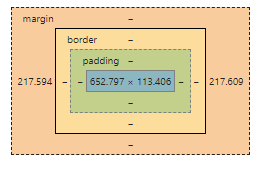
1. title



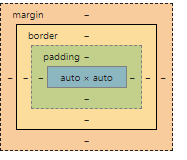
1. back



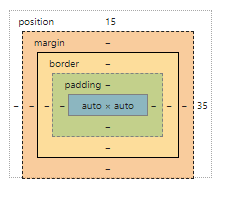
1. poster
   1. 
2. myImg



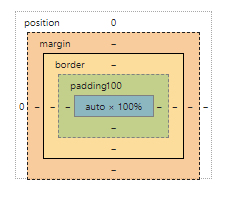
1. menu

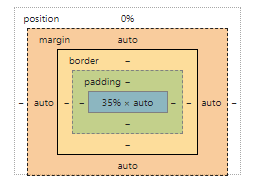


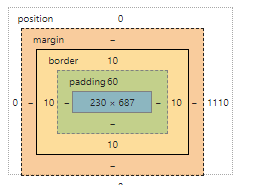
1. close2



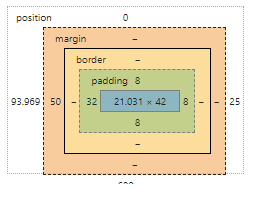
1. myModal



1. Modal\_Content
   1. 
2. Menu



1. close



### Code

<!DOCTYPE html>

<html lang="en"> ***<!--set up that this is an html file and the language is english-->***

<head>

<meta charset="UTF-8"> ***<!--defines the type of charcters being used-->***

<style>

/\*sets settings and background\*/

\* { ***/\*all sellector\*/***

box-sizing: border-box; ***/\*The CSS box-sizing property allows us to include the padding and border in an element’s total width and height. The border box variable makes it so element padding and border are included in the width and height\*/***

}

body {

background-image: url("bgtile.jpg"); ***/\*sets the background image\*/***

}

.Main\_Content { ***/\*selector for main content area\*/***

position: absolute; /***\*The position property specifies the type of positioning method used for an element. The absolute variable means The element is positioned relative to anything it is containted within. In this case the body. \*/***

width:80%; ***/\*sets the width to 80%\*/***

height:80%; /***\*sets the height to 80%\*/***

right:10%; ***/\*put***s ***it 10% from the right\*/***

top:10%; ***/\*sets it 10% from the top\*/***

background-color: white;

}

***/\*CSS for the side menu bar\*/***

***/\* Set up the side bar menu itself. Javascript will make it so that it will appear by chaning the width from zero \*/***

.Menu { ***/\*selects the overall menu section\*/***

height: 100%; /***\*sets the height to 100%\*/***

width: 0; ***/\*changed in javascript when you click to open the menu\*/***

position: fixed; /***\*The position property specifies the type of positioning method used for an element. The fixed variable means that the element is positioned relative to the browser window\*/***

z-index: 4; ***/\*The z-index property specifies the stack order of an element. An element with greater stack order is always in front of an element with a lower stack order.\*/***

top: 0***; /\*puts it at the top\*/***

left: 0; /\****puts it to the left\*/***

background-color: #fafafa;

overflow-x: hidden; ***/\*The overflow-x property specifies whether to clip the content, add a scroll bar, or display overflow content of a block-level element, when it overflows at the left and right edges. The hidden variable makes it so the content is clipped - and no scrolling mechanism is provided\*/***

transition: 0.2s; ***/\*Specifies how many seconds or milliseconds the transition effect takes to complete. In this case the transition form 0 width to full side bar width\*/***

padding-top: 60px; ***: /\*Padding creates extra space within an element on the top\*/***

}

***/\*Design for the words within the side bar menu\*/***

.Menu a { ***/\*selects the words within the side bar menu\*/***

padding: 8px 8px 8px 32px; ***/\* sets extra space within the element in the order of top, right, bottom, then left\*/***

text-decoration: none; ***/\*takes away the standard underline in a link\*/***

font-size: 25px;

font-family: Arial***;/\*sets the font\*/***

color: black;

display: block; ***/\*The display property specifies the display behavior (the type of rendering box) of an element. The block variable means it displays an element as a block element (like <p>). It starts on a new line, and takes up the whole width\*/***

transition: 0.1s; ***/\*Specifies how many seconds or milliseconds the transition effect takes to complete. In this case the transition form 0 width to full side bar width\*/***

}

***/\*Positions and designs the menu close button\*/***

.Menu .close { /***\*selects the close icon within the menu sidebar\*/***

position: absolute; ***/\*The position property specifies the type of positioning method used for an element. The absolute variable means The element is positioned relative to anything it is containted within. In this case the sidebar menu. \*/***

top: 0; ***/\*puts it at the top\*/***

right: 25px; ***/\*puts it that amount from the right\*/***

font-size: 36px;

margin-left: 50px; ***/\*margin creates extra space around an element in this case to the left\*/***

}

***/\*CSS for the images shown when you first see the page\*/***

#myImg { ***/\*calls on anthing with the id= set it myIMG in this case this is the four impact headers to click on\*/***

cursor: pointer; ***/\*sets the cursor to the pointer hand while cursor over it\*/***

transition: 0.3s; ***/\*Specifies how many seconds or milliseconds the transition*** ***effect takes to complete. In this case the transition from as it is to hover settings\*/***

display: block; ***/\*The display property specifies the display behavior (the type of rendering box) of an element. The block variable means it displays an element as a block element (like <p>). It starts on a new line, and takes up the whole width\*/***

margin-left: auto; ***/\*effectly this in combination with the dispaly setting centers an element\*/***

margin-right: auto;

width: 40%; ***/\*sets the width to 40%\*/***

}

#myImg:hover {opacity: 0.7;}***/\*this makes the element semi-transparent when the mouse is hovering over it\*/***

***/\*CSS for the modal or the pop up window when you click on a button\*/***

.modal { ***/\*the element that pops up after you click\*/***

display: none; ***/\* Hidden by default \*/***

position: fixed; ***/\*The position property specifies the type of positioning method used for an element. The fixed variable means that the element is positioned relative to the browser window\*/***

z-index: 4; ***/\*The z-index property specifies the stack order of an element. An element with greater stack order is always in front of an element with a lower stack order.\*/***

padding-top: 100px; ***/\*Padding creates extra space within an element on the top\*/***

***left: 0; /\*puts the element all the way to the left\*/***

top: 0; ***/puts the element all the way to the top\*/***

width: 100%; ***/\* Full width \*/***

height: 100%; ***/\* Full height \*/***

overflow: auto; ***/\* Enable scroll if needed \*/***

background-color: rgb(0,0,0); ***/\* Fallback color \*/***

background-color: rgba(0,0,0,0.9); ***/\* Black w/ opacity \*/***

}

***/\* CSS for themodal content or the image showed in the pop up \*/***

.modal-content {

position:relative***;/\*The position property specifies the type of positioning method used for an element. The variable realtive means the element is positioned relative to its normal position, so "left:20px" adds 20 pixels to the element's LEFT position\*/***

top:-0%; ***/\*Makes it all the way to the top\*/***

margin: auto; ***/\*effectly this in combination with the dispaly setting centers an element\*/***

display: block; ***/\*effectly this in combination with the dispaly setting centers an element\*/***

width: 35%; ***/\*sets the width to 35%\*/***

}

***/\* Animated the popup window \*/***

.modal-content {

-webkit-animation-name: zoom; ***/\*property specifies a name for the @keyframes animation.\*/***

-webkit-animation-duration: 0.6s; ***/\*property defines how long an animation should take to complete one cycle.\*/***

animation-name: zoom; ***/\*property specifies a name for the @keyframes animation.\*/***

animation-duration: 0.6s; ***/\*property defines how long an animation should take to complete one cycle.\*/***

}

@-webkit-keyframes zoom {

from {-webkit-transform:scale(0)} ***/\*goes from a scale of nothing\*/***

to {-webkit-transform:scale(1)} ***/\*to full\*/***

}

@keyframes zoom {

from {transform:scale(0)} ***/\*goes from a scale of nothing\*/***

to {transform:scale(1)} ***/\*to full\*/***

}

***/\* CSS for the close button \*/***

.close2 { ***/\*selects the close button for the popup window\*/***

position: absolute; ***/\*The position property specifies the type of positioning method used for an element. The absolute variable means The element is positioned relative to anything it is containted within. In this case the body. \*/***

top: 15px; ***/\*puts it 15 pixels from the top\*/***

right: 35px; /***\*puts it 35 pixels from the right\*/***

color: #f1f1f1;

font-size: 100px;

font-weight: bold***;/\*makes the font bold just like in word\*/***

transition: 0.3s; ***/\*CSS transitions allows you to change property values smoothly*** ***(from one value to another), over a given duration.\*/***

cursor:pointer; ***/\*sets the cursor to the pointer hand while cursor over it\*/***

}

#title{ ***/\*selects the header\*/***

text-align: center; ***/\*The text-align property specifies the horizontal alignment of text in an element. In this case it is to the center\*/***

font-size: 60px;

color: white;

background-color:#000066;

position:relative; ***/\*The position property specifies the type of positioning method used for an element. The variable realtive means the element is positioned relative to its normal position, so "left:20px" adds 20 pixels to the element's LEFT position\*/***

top:-10%; ***/\* it is moved up relative to where it used to be\*/***

font-family: Arial Black; ***/\*sets the font\*/***

z-index: 1; ***/\*The z-index property specifies the stack order of an element. An element with greater stack order is always in front of an element with a lower stack order.\*/***

}

.poster{ /***\*selects all the images auto displayed on opening\*/***

position:relative; ***/\*The position property specifies the type of positioning method used for an element. The variable realtive means the element is positioned relative to its normal position, so "left:20px" adds 20 pixels to the element's LEFT position\*/***

top:-10%; ***/\* it is moved up relative to where it used to be\*/***

}

.back{ ***/\*selects the character of the back button\*/***

z-index: 3; ***/\*The z-index property specifies the stack order of an element. An element with greater stack order is always in front of an element with a lower stack order.\*/***

position: absolute; ***/\*The position property specifies the type of positioning method used for an element. The absolute variable means The element is positioned relative to anything it is containted within. In this case the body. \*/***

top:-5%; ***/\* it is moved up relative to where it used to be\*/***

color: white;

width:50px***;/\*sets the width to 50 pixels\*/***

height:10%; ***/\*sets the height to 10%\*/***

left:1%; ***/\*it is moved a little form the left\*/***

font-family: Arial; ***/\*sets the font\*/***

}

.backBG{ ***/\*selects the baclground of the back button\*/***

height: 85px; /***\*it has a height of 85 pixels\*/***

width: 85px; ***/\*it has a width of 85 pixels\*/***

background-color: #4d6e8f;

z-index: 2; ***/\*The z-index property specifies the stack order of an element. An element with greater stack order is always in front of an element with a lower stack order.\*/***

position: absolute; ***/\*The position property specifies the type of positioning method used for an element. The absolute variable means The element is positioned relative to anything it is containted within. In this case the sidebar menu. \*/***

top:-3.4%; ***/\* it is moved up relative to where it used to be\*/***

left:0%; ***/\*it is all the way to the left\*/***

opacity: 0.8***;/\*makes it semi-tranparent\*/***

}

</style>

</head>

<body>

<div id="Menu" class="Menu"> ***<!--sets up the container for the side bar menu-->***

<a href="javascript:void(0)" class="close" onclick="closeMenu()">&times;</a> ***<!--created the close icon that on click actiacted the closeMenu() function. The symbol is create using a char code-->***

<a href="Team.html">Blue Cheese Robotics</a> ***<!--makes the words take the user to the defind href="" location when clicked-->***

<a href="Outreach.html">Our Outreach</a> ***<!--makes the words take the user to the defind href="" location when clicked-->***

<a href="Competition.html">Game Manual</a> ***<!--makes the words take the user to the defind href="" location when clicked-->***

</div>

<span style="font-size:45px;cursor:pointer;color:white;font-family: Arial" onclick="openMenu()">&#9776; Menu</span> ***<!--styles the open menu icon including making the cursor a pointer while above the element, sets the font, and calls the function openMenu() to open the menu when clicked. The wierd charcers infornt of menu makes the three bars symbol.-->***

<script> ***<!--defines when javascript starts-->***

function openMenu() { ***//function called when the user clicks on the span with the menu in the corner***

document.getElementById("Menu").style.width = "250px"; //***changes the width to make the side bar slide out assisted with the transition in css***

document.getElementById("Menu").style.border = "10px solid #b3b3ff"; ***//makes the border around the element come with the sidebar***

***}***

function closeMenu() {

document.getElementById("Menu").style.width = "0"; ***<!--changes the width to make the side bar slide in assisted with the transition in css-->***

document.getElementById("Menu").style.border = "0px"; ***<!--makes it so the border goes away when the side bar goes away-->***

}

</script> ***<!--defines when javascript ends-->***

<div class= "Main\_Content"> ***<!-- container for the actual content of the page-->***

<a href="FIRST.html"> ***<!--makes everythign inside a link to href=""-->***

<span class="back" style='font-size:85px;font-family: Arial;'>&#8249;&#8249;</span>***<!-=creates the back button-->***

<span class="backBG"></span> ***<!--makes the background for the back button-->***

</a>

<h1 id="title">Impact</h1> ***<!--creates the header at the top-->***

<div class= "poster"> ***<!--creates the container to hold all the elements that make up the diplayed images-->***

<img id="myImg" onclick="openImg('a')" src="Impact/poster1a.jpg" style="width:60%"> ***<!-Creates individual images you first see and calls a function when they are clicked-->***

<img id="myImg" onclick="openImg1('a')" src="Impact/poster2a.jpg" style="width:60%">

<img id="myImg" onclick="openImg2('a')" src="Impact/poster3a.jpg" style="width:60%">

<img id="myImg" onclick="openImg3('a')" src="Impact/poster4a.jpg" style="width:60%">

</div>

<div id="myModal" class="modal"> ***<!--creates the container that contains the pop up-->***

<span class="close2" onclick="closeImg()">&times;</span> ***<!-- creates the close button that calls a function to close the popup-->***

<img class="modal-content" id="img01"> ***<!-- creates and img container that will have the src="" set in javascript when the popup is opened-->***

</div>

<script>

var modal = document.getElementById('myModal'); ***// Get the modal*** variable

function openImg(num){ ***// start of function***

if(num = 'a'){

modal.style.display = "block"; //***when called chaages the diplay to block or more so displays an element as a block element (like <p>). It starts on a new line, and takes up the whole width instead of none***

document.getElementById("img01").src = "Impact/poster1b.jpg"; ***//insert the img called for when click un the button***

}

}

function openImg1(num){ ***// start of function***

if(num = 'a'){

modal.style.display = "block"; ***//when called chaages the diplay to block or more so displays an element as a block element (like <p>). It starts on a new line, and takes up the whole width instead of none***

document.getElementById("img01").src = "Impact/poster2b.jpg"; ***//insert the img called for when click un the button***

}

}

function openImg2(num){ ***// start of function***

if(num = 'a'){

modal.style.display = "block"; ***//when called chaages the diplay to block or more so displays an element as a block element (like <p>). It starts on a new line, and takes up the whole width instead of none***

document.getElementById("img01").src = "Impact/poster3b.jpg"; ***//insert the img called for when click un the button***

}

}

function openImg3(num){ ***// start of function***

if(num = 'a'){

modal.style.display = "block"; ***//when called chaages the diplay to block or more so displays an element as a block element (like <p>). It starts on a new line, and takes up the whole width instead of none***

document.getElementById("img01").src = "Impact/poster4b.jpg"; ***//insert the img called for when click un the button***

}

}

function closeImg(){ ***// start of function***

modal.style.display = "none"; ***//when called changes the diplay setting back to none effectvly making it go away***

}

</script>

</div>

</body>

</html

## Levels

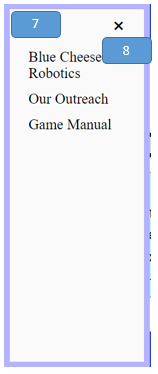
### Overview

Content: Interactive display of the four levels of FIRST with tabs for each level that display content.

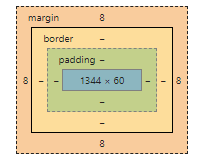
Purpose: To give information of the different levels of FIRST and to help other sees in which ones they can get involved by telling age groups and cost.

Audience: Users with little to none prior experience with FIRST.

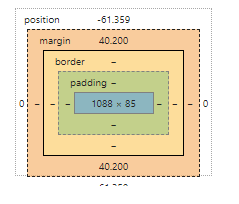
### Layout



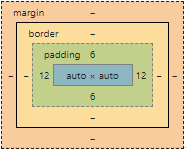
1. body



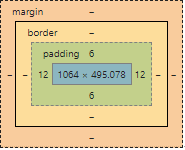
1. title



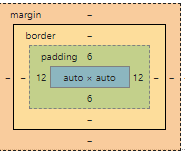
1. tabcontent
   1. FRC



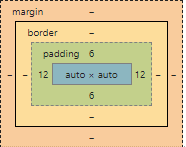
* 1. FLL



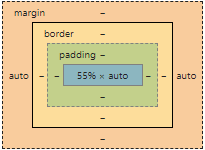
* 1. FLL JR



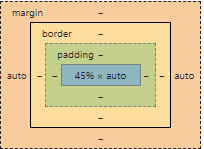
* 1. FTC



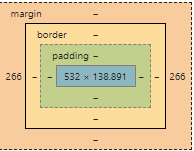
1. Main\_Image
   1. FRC



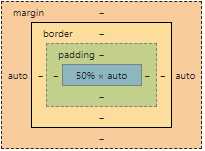
* 1. FLL



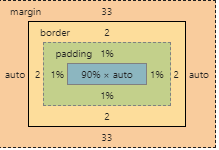
* 1. FLL Jr



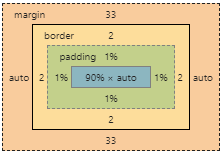
* 1. FTC



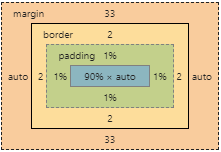
1. p
   1. FRC



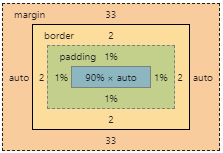
* 1. FLL



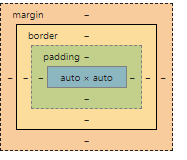
* 1. FLLJR



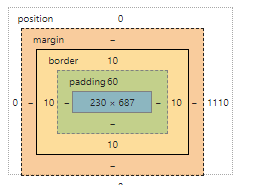
* 1. FTC



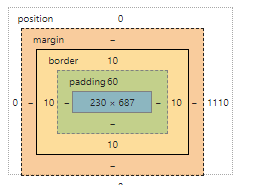
1. menu



1. Menu



1. close



### Code

<!DOCTYPE html>

<html lang="en"> ***<!--set up that this is an html file and the language is english-->***

<head>

<meta charset="UTF-8***"> <!--defines the type of charcters being used-->***

<style>

***/\*sets settings and background\*/***

\* { ***/\*all sellector\*/***

box-sizing: border-box;

}

body {

background-image: url("bgtile.jpg"); ***/\*sets the background image\*/***

}

.Main\_Content { ***/\*selector for main content area\*/***

position: absolute; ***/\*The position property specifies the type of positioning method used for an element. The absolute variable means The element is positioned relative to anything it is containted within. In this case the body. \*/***

width:80%; ***/\*sets the width to 80%\*/***

height:80%; /***\*sets the height to 80%\*/***

right:10%; ***/\*puts it 10% from the right\*/***

top:10%; ***/\*sets it 10% from the top\*/***

background-color: white;

}

.Main\_Image { ***/\*selector for image\*/***

display: block; ***/\*The display property specifies the display behavior (the type of rendering box) of an element. The block variable means it displays an element as a block element (like <p>). It starts on a new line, and takes up the whole width\*/***

margin-left: auto; ***/\*effectly this in combination with the dispaly setting centers an element\*/***

margin-right: auto;

width: 60%; /\*sets width to 60%\*/

}

***/\*CSS for the side menu bar\*/***

***/\* Set up the side bar menu itself. Javascript will make it so that it will appear by chaning the width from zero \*/***

.Menu { ***/\*selects the overall menu section\*/***

height: 100%; /\****sets the height to 100%\*/***

width: 0; ***/\*changed in javascript when you click to open the menu\*/***

position: fixed; ***/\*The position property specifies the type of positioning method used for an element. The fixed variable means that the element is positioned relative to the browser window\*/***

z-index: 4; ***/\*The z-index property specifies the stack order of an element. An element with greater stack order is always in front of an element with a lower stack order.\*/***

top: 0; ***/\*puts it at the top\*/***

left: 0; ***/\*puts it to the left\*/***

background-color: #fafafa;

overflow-x: hidden; ***/\*The overflow-x property specifies whether to clip the content, add a scroll bar, or display overflow content of a block-level element, when it overflows at the left and right edges. The hidden variable makes it so the content is clipped - and no scrolling mechanism is provided\*/***

transition: 0.2s; ***/\*Specifies how many seconds or milliseconds the transition effect takes to complete. In this case the transition form 0 width to full side bar width\*/***

padding-top: 60px; : ***/\*Padding creates extra space within an element on the top\*/***

}

***/\*Design for the words within the side bar menu\*/***

.Menu a { ***/\*selects the words within the side bar menu\*/***

padding: 8px 8px 8px 32px; ***/\* sets extra space within the element in the order of top, right, bottom, then left\*/***

text-decoration: none; ***/\*takes away the standard underline in a link\*/***

font-size: 25px;

color: black;

font-family: Arial;/***\*sets the font\*/***

display: block; ***/\*The display property specifies the display behavior (the type of rendering box) of an element. The block variable means it displays an element as a block element (like <p>). It starts on a new line, and takes up the whole width\*/***

transition: 0.1s; ***/\*Specifies how many seconds or milliseconds the transition effect takes to complete. In this case the transition form 0 width to full side bar width\*/***

}

***/\*Positions and designs the menu close button\*/***

.Menu .close { ***/\*selects the close icon within the menu sidebar\*/***

position: absolute; ***/\*The position property specifies the type of positioning method used for an element. The absolute variable means The element is positioned relative to anything it is containted within. In this case the sidebar menu. \*/***

top: 0; ***/\*puts it at the top\*/***

right: 25px; ***/\*puts it that amount from the right\*/***

font-size: 36px;

margin-left: 50px; ***/\*margin creates extra space around an element in this case to the left\*/***

}

.tab { ***/\*selects the buttons appreaing at the top of the page\*/***

overflow: hidden; ***/\*The overflow-x property specifies whether to clip the content, add a scroll bar, or display overflow content of a block-level element, when it overflows at the left and right edges. The hidden variable makes it so the content is clipped - and no scrolling mechanism is provided\*/***

border: 1px solid #296eb4; ***/\*this creates a border around the element it is define by width, what type, then color\*/***

border-style: none solid none solid; ***/\*this defined the border on each side in the order of top, right, bottom, then left\*/***

background-color: #296eb4;

position:relative; ***/\*The position property specifies the type of positioning method used for an element. The variable realtive means the element is positioned relative to its normal position, so "left:20px" adds 20 pixels to the element's LEFT position\*/***

top:-17.2%; ***\*makes it higher than it would usually be\*/***

}

.tab button { ***/\*selects the button part that acts as a button of the tabs at the top of the page\*/***

background-color: inherit;/***\*same background color as tab as it is it's parent element or elmenet that it is within\*/***

float: left;

cursor: pointer; ***/\*sets the cursor to the pointer hand while cursor over it\*/***

padding: 14px 16px;

transition: 0.3s;

font-size: 17px;

border: 1px solid #ccc; ***/\*this creates a border around the element it is define by width, what type, then color\*/***

width:25%; /***\*this sets the width to 25%\*/***

color: white;

font-family: "Lucida Sans Unicode"; /***\*sets the font\*/***

}

.tab button.active { ***/\*choose the tab that is in use \*/***

background-color: #5b9dd7;

}

.tabcontent { /***\*selects the content displayed after clicking the buttons\*/***

display: none; ***/\*Hides content until javascript changes\*/***

padding: 6px 12px;

text-indent: 50px; ***/\*The text-indent property specifies the indentation of the first line in a text-block.\*/***

text-align: justify; ***/\*The text-align property specifies the horizontal alignment of text in an element. The jusify variable means it stretches the lines so that each line has equal width (like in newspapers and magazines)\*/***

position:relative; ***/\*The position property specifies the type of positioning method used for an element. The variable realtive means the element is positioned relative to its normal position, so "left:20px" adds 20 pixels to the element's LEFT position\*/***

top:-15%; ***/\* it is moved up relative to where it used to be\*/***

}

#title{ ***/\*selects the header\*/***

text-align: center; ***/\*The text-align property specifies the horizontal alignment of text in an element. In this case it is to the center\*/***

font-size: 60px;

color: white;

background-color:#000066;

position:relative; ***/\*The position property specifies the type of positioning method used for an element. The variable realtive means the element is positioned relative to its normal position, so "left:20px" adds 20 pixels to the element's LEFT position\*/***

top:-10%; ***/\* it is moved up relative to where it used to be\*/***

font-family: Arial Black; ***/\*sets the font\*/***

z-index: 1; ***/\*The z-index property specifies the stack order of an element. An element with greater stack order is always in front of an element with a lower stack order.\*/***

}

p{

text-indent: 50px; ***/\*The text-indent property specifies the indentation of the first line in a text-block.\*/***

text-align: justify***; /\*The text-align property specifies the horizontal alignment of text in an element. The jusify variable means it stretches the lines so that each line has equal width (like in newspapers and magazines)\*/***

border: 2px solid #9d9daf; ***/\*this creates a border around the element it is define by width, what type, then color\*/***

border-radius: 8px; ***/\*this rounds the border around an element\*/***

padding: 1%; ***/\*Padding creates extra space within an element\*/***

display: block; ***/\*The display property specifies the display behavior (the type of rendering box) of an element. The block variable means it displays an element as a block element (like <p>). It starts on a new line, and takes up the whole width\*/***

margin-left: auto; ***/\*effectly this in combination with the dispaly setting centers an element\*/***

margin-right: auto;

font-size:33px;

font-family: "Lucida Sans Unicode"; ***/\*sets the font\*/***

}

.back{ ***/\*selects the character of the back button\*/***

z-index: 3; ***/\*The z-index property specifies the stack order of an element. An element with greater stack order is always in front of an element with a lower stack order.\*/***

position: absolute; /***\*The position property specifies the type of positioning*** ***method used for an element. The absolute variable means The element is positioned relative to anything it is containted within. In this case the body. \*/***

top:-5%; ***/\* it is moved up relative to where it used to be\*/***

color: white;

width:50px;***/\*sets the width to 50 pixels\*/***

height:10%; ***/\*sets the height to 10%\*/***

left:1%; ***/\*it is moved a little form the left\*/***

font-family: Arial; ***/\*sets the font\*/***

}

.backBG{ ***/\*selects the baclground of the back button\*/***

height: 85px; ***/\*it has a height of 85 pixels\*/***

width: 85px; ***/\*it has a width of 85 pixels\*/***

background-color: #4d6e8f;

z-index: 2; ***/\*The z-index property specifies the stack order of an element. An element with greater stack order is always in front of an element with a lower stack order.\*/***

position: absolute; ***/\*The position property specifies the type of positioning method used for an element. The absolute variable means The element is positioned relative to anything it is containted within. In this case the sidebar menu. \*/***

top:-3.4%; /***\* it is moved up relative to where it used to be\*/***

left:0%; ***/\*it is all the way to the left\*/***

opacity: 0.8;***/\*makes it semi-tranparent\*/***

}

</style>

</head>

<body>

<div id="Menu" class="Menu"> ***<!--sets up the container for the side bar menu-->***

<a href="javascript:void(0)" class="close" onclick="closeMenu()">&times;</a> ***<!--created the close icon that on click actiacted the closeMenu() function. The symbol is create using a char code-->***

<a href="Team.html">Blue Cheese Robotics</a> ***<!--makes the words take the user to the defind href="" location when clicked-->***

<a href="Outreach.html">Our Outreach</a> ***<!--makes the words take the user to the defind href="" location when clicked-->***

<a href="Competition.html">Game Manual</a> ***<!--makes the words take the user to the defind href="" location when clicked-->***

</div>

<span style="font-size:45px;cursor:pointer;color:white;font-family: Arial" onclick="openMenu()">&#9776; Menu</span> ***<!--styles the open menu icon including making the cursor a pointer while above the element, sets the font, and calls the function openMenu() to open the menu when clicked. The wierd charcers infornt of menu makes the three bars symbol.-->***

<script> <!--defines when javascript starts-->

function openMenu() { ***//function called when the user clicks on the span with the menu in the corner***

document.getElementById("Menu").style.width = "250px"; ***//changes the width to make the side bar slide out assisted with the transition in css***

document.getElementById("Menu").style.border = "10px solid #b3b3ff"; ***//makes the border around the element come with the sidebar***

}

function closeMenu() {

document.getElementById("Menu").style.width = "0"; ***//changes the width to make the side bar slide in assisted with the transition in css***

document.getElementById("Menu").style.border = "0px"; ***//makes it so the border goes away when the side bar goes away***

}

</script> ***<!--defines when javascript ends-->***

<div class= "Main\_Content"> ***<!-- container for the actual content of the page-->***

<a href="FIRST.html"> ***<!--makes everythign inside a link to href=""-->***

<span class="back"style='font-size:85px;font-family: Arial;'>&#8249;&#8249;</span>***<!-=creates the back button-->***

<span class="backBG"></span> ***<!--makes the background for the back button-->***

</a>

<h1 id="title">Levels</h1> ***<!--creates the header at the top-->***

<div class="tab"> ***<!--creates the tab section at the top-->***

<button class="tablinks" onclick="openNow(event, 'FIRST Robotics Competition')">FIRST Robotics Competition</button> ***<!-- creates a button for each level that calls a function and based on the variable passed through opens a new tab contnent-->***

<button class="tablinks" onclick="openNow(event, 'FIRST LEGO League')">FIRST LEGO League</button>

<button class="tablinks" onclick="openNow(event, 'FIRST LEGO League Junior')">First LEGO League Junior</button>

<button class="tablinks" onclick="openNow(event, 'FIRST Tech Challenge')">First Tech Challenge</button>

</div>

<div id="FIRST Robotics Competition" class="tabcontent"> ***<!--holds a set of tab content for a corresponding tab-->***

<img class= "Main\_Image" style="width:55%" src="Levels/FRCVECTOR.svg"> ***<!--hold the image for in the tab content-->***

<p style="background-color:#cceeff;color:black;"> In the FIRST Robotics Competition, High school students on teams typically ranging from thirty to fifty members work with adult coaches and mentors to design, build, and program a robot to compete in a specific game, announced each January. </p>

***<!--creates the written content to go in the tab content along with setting it's color scheme-->***

</div>

<div id="FIRST LEGO League" class="tabcontent"> ***<!--holds a set of tab content for a corresponding tab-->***

<img class= "Main\_Image" style="width:45%" src= "Levels/FLLVECTOR.svg"> ***<!--hold the image for in the tab content-->***

<p style="background-color:#cceeff;color:black;" >FIRST LEGO League is a program created to inspire fourth through eighth graders through designing a robot and programming it to automatically carry out missions on a table and by researching a problem and crafting a creative solution. </p>

***<!--creates the written content to go in the tab content along with setting it's color scheme-->***

</div>

<div id="FIRST LEGO League Junior" class="tabcontent"> ***<!--holds a set of tab content for a corresponding tab-->***

<img class= "Main\_Image" style="width:50%" src= "Levels/FLLJRVECTOR.svg"> ***<!--hold the image for in the tab content-->***

<p style="background-color:#cceeff;color:black;">FIRST LEGO League Jr. is a program created for elementary school students aged 6-10 in which a team solves a real-world challenge by applying research and creativity to the problem.</p>

***<!--creates the written content to go in the tab content along with setting it's color scheme-->***

</div>

<div id="FIRST Tech Challenge" class="tabcontent"> ***<!--holds a set of tab content for a corresponding tab-->***

<img class="Main\_Image" style="width:50%" src="Levels/FTCVECTOR.svg"> ***<!--hold the image for in the tab content-->***

<p style="background-color:#cceeff;color:black;">The FIRST Tech Challenge is a scaled-down version of the FIRST Robotics Competition that is open to grades 7-12 where teams are challenged to design, build, program, and operate robots to compete. </p>

***<!--creates the written content to go in the tab content along with setting it's color scheme-->***

</div>

<script>

function openNow(evt, dtls) { ***//is function callee when the click on a button that takes in information about which buttton it is in order to show the right thing***

var i, tabcontent, tablinks; ***//creates or initializes these variables***

tabcontent = document.getElementsByClassName("tabcontent");/***/sets the variable tabcontnet to the tabcontent we have already made in html***

for (i = 0; i < tabcontent.length; i++) { ***//goes through and make the display none again for all***

tabcontent[i].style.display = "none";

}

tablinks = document.getElementsByClassName("tablinks"); ***//get the tablinks in the tab and makes them*** ***active if they are not already in order to chnage the color***

for (i = 0; i < tablinks.length; i++) {

tablinks[i].className = tablinks[i].className.replace(" active", "");

}

document.getElementById(dtls).style.display = "block"; ***//makes the one has been clicked on dispaly.***

evt.currentTarget.className += " active";***//marks the only one supposed to be active as active***

}

</script>

</body>

</html>

## Team

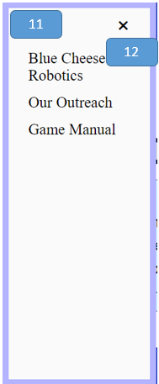
### Overview

Content: Start page that contains links to the subpages for content about blue cheese.

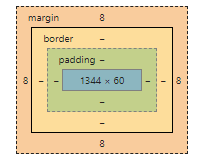
Purpose: Into to the section about the team, Blue Cheese Robotics.

Audience: Non-blue cheese affiliated users.

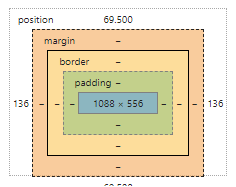
### Layout



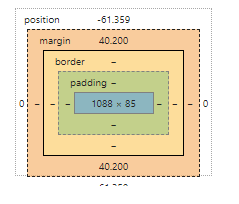
1. body



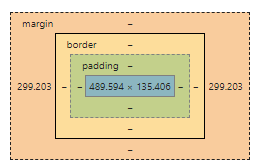
1. Main\_Content



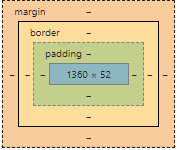
1. title



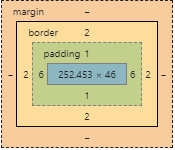
1. Main\_Image



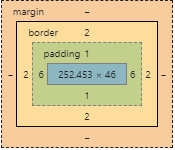
1. buttons



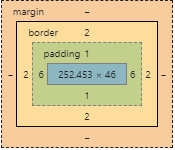
1. button1



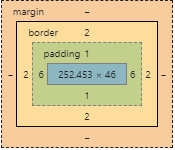
1. buttton2



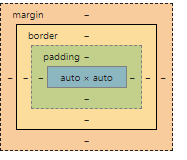
1. button3



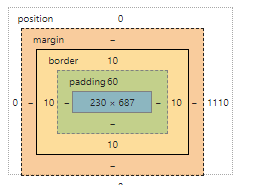
1. buttton4



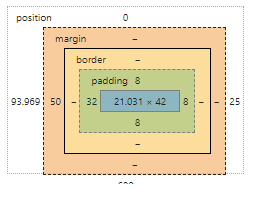
1. menu



1. Menu



1. close



### Code

<!DOCTYPE html>

<html lang="en"> ***<!--set up that this is an html file and the language is english-->***

<head>

<meta charset="UTF-8"> ***<!--defines the type of charcters being used-->***

<style>

***/\*sets settings and background\*/***

\* { ***/\*all sellector\*/***

box-sizing: border-box; ***/\*The CSS box-sizing property allows us to include the padding and border in an element’s total width and height. The border box variable makes it so element padding and border are included in the width and height\*/***

}

body { ***/\*effectivly all selector\*/***

background-image: url("bgtile.jpg"); ***/\*sets the background image\*/***

}

/\****CSS for the main content area\*/***

.Main\_Content { /***\*selector for main content area\*/***

position: absolute***; /\*The position property specifies the type of positioning method used for an element. The absolute variable means The element is positioned relative to anything it is containted within. In this case the body. \*/***

width:80%; ***/\*sets the width to 80%\*/***

height:80%; ***/\*sets the height to 80%\*/***

right:10%; ***/\*puts it 10% from the right\*/***

top:10%; ***/\*sets it 10% from the top\*/***

background-color: white;

}

.Main\_Image { ***/\*selects the central image or more so gif displayed\*/***

border: 7px solid #fee622; ***/\*this creates a border around the element it is define by width, what type, then color\*/***

width: 45%; ***/\*sets the width to 45%\*/***

top:30%; ***/\*moves it 30% from the top\*/***

display: block; ***/\*The display property specifies the display behavior (the type of rendering box) of an element. The block variable means it displays an element as a block element (like <p>). It starts on a new line, and takes up the whole width\*/***

margin-left: auto***;/\*effectly this in combination with the dispaly setting centers an element\*/***

margin-right: auto;

}

.Main\_Content .buttons{ ***/\*selector that positions buttons in terms of being withing the main content\*/***

position:fixed***;/\*The position property specifies the type of positioning method used for an element. The fixed variable means that the element is positioned relative to the browser window\*/***

left:10%; /***\*moves it 10% from the left\*/***

bottom:15%; ***/\*moves it 15% from the bottom\*/***

}

button { ***/\*selects all buttons\*/***

background-color: black;

color: white;

cursor: pointer; ***/\*sets the cursor to the pointer hand while cursor over it\*/***

font-size: 30px;

font-family: "Lucida Sans Unicode"; /***\*sets the font\*/***

box-sizing: border-box; ***/\*The CSS box-sizing property allows us to include the padding and border in an element’s total width and height. The border box variable makes it so element padding and border are included in the width and height\*/***

}

***/\*CSS for the side menu bar\*/***

***/\* Set up the side bar menu itself. Javascript will make it so that it will appear by chaning the width from zero \*/***

.Menu { ***/\*selects the overall menu section\*/***

height: 100%; ***/\*sets the height to 100%\*/***

width: 0; ***/\*changed in javascript when you click to open the menu\*/***

position: fixed; ***/\*The position property specifies the type of positioning method used for an element. The fixed variable means that the element is positioned relative to the browser window\*/***

z-index: 4; ***/\*The z-index property specifies the stack order of an element. An element with greater stack order is always in front of an element with a lower stack order.\*/***

top: 0; ***/\*puts it at the top\*/***

left: 0; ***/\*puts it to the left\*/***

background-color: #fafafa;

overflow-x: hidden; ***/\*The overflow-x property specifies whether to clip the content, add a scroll bar, or display overflow content of a block-level element, when it overflows at the left and right edges. The hidden variable makes it so the content is clipped - and no scrolling mechanism is provided\*/***

transition: 0.2s; ***/\*Specifies how many seconds or milliseconds the transition effect takes to complete. In this case the transition form 0 width to full side bar width\*/***

padding-top: 60px; : /***\*Padding creates extra space within an element on the top\*/***

}

***/\*Design for the words within the side bar menu\*/***

.Menu a { ***/\*selects the words within the side bar menu\*/***

padding: 8px 8px 8px 32px; ***/\* sets extra space within the element in the order of top, right, bottom, then left\*/***

text-decoration: none; ***/\*takes away the standard underline in a link\*/***

font-size: 25px;

font-family: Arial;/***\*sets the font\*/***

color: black;

display: block; ***/\*The display property specifies the display behavior (the type of rendering box) of an element. The block variable means it displays an element as a block element (like <p>). It starts on a new line, and takes up the whole width\*/***

transition: 0.1s; ***/\*Specifies how many seconds or milliseconds the transition effect takes to complete. In this case the transition form 0 width to full side bar width\*/***

}

***/\*Positions and designs the menu close button\*/***

.Menu .close { ***/\*selects the close icon within the menu sidebar\*/***

position: absolute; ***/\*The position property specifies the type of positioning method used for an element. The absolute variable means The element is positioned relative to anything it is containted within. In this case the sidebar menu. \*/***

top: 0; ***/\*puts it at the top\*/***

right: 25px; ***/\*puts it that amount from the right\*/***

font-size: 36px;

margin-left: 50px; ***/\*margin creates extra space around an element in this case to the left\*/***

}

.content1 {

position:relative; ***/\*The position property specifies the type of positioning method used for an element. The variable realtive means the element is positioned relative to its normal position, so "left:20px" adds 20 pixels to the element's LEFT position\*/***

top:7%;***/\*moves it 7% from the top\*/***

width:50%; ***/\*sets the width to 50%\*/***

display: block; ***/\*The display property specifies the display behavior (the type of rendering box) of an element. The block variable means it displays an element as a block element (like <p>). It starts on a new line, and takes up the whole width\*/***

margin-left: auto***;/\*effectly this in combination with the dispaly setting centers an element\*/***

margin-right: auto;

padding: 1%; ***/\*Padding creates extra space within an element\*/***

}

.header{

background-color:#011298;

width:100%; ***/\*sets the width to 100%\*/***

}

.content2 {

text-indent: 50px; ***/\*The text-indent property specifies the indentation of the first line in a text-block.\*/***

text-align: justify; ***/\*The text-align property specifies the horizontal alignment of text in an element. The jusify variable means it stretches the lines so that each line has equal width (like in newspapers and magazines)\*/***

border: 5px solid #b3b3ff; ***/\*this creates a border around the element it is define by width, what type, then color\*/***

border-radius: 8px; ***/\*this rounds the border around an element\*/***

padding: 1%; ***/\*Padding creates extra space within an element\*/***

width: 90%; /***\*sets the width to 90%\*/***

display: block; ***/\*The display property specifies the display behavior (the type of rendering box) of an element. The block variable means it displays an element as a block element (like <p>). It starts on a new line, and takes up the whole width\*/***

margin-left: auto; ***/\*effectly this in combination with the dispaly setting centers an element\*/***

margin-right: auto;

font-size:20px;

font-family: "Lucida Sans Unicode", "Lucida Grande", sans-serif***;/\*sets the font\*/***

}

</style>

</head>

<body>

<div id="Menu" class="Menu"> ***<!--sets up the container for the side bar menu-->***

<a href="javascript:void(0)" class="close" onclick="closeMenu()">&times;</a> ***<!-created the close icon that on click actiacted the closeMenu() function. The symbol is created using a char code-->***

<a href="Team.html">Blue Cheese Robotics</a> ***<!--makes the words take the user to the defind href="" location when clicked-->***

<a href="Outreach.html">Our Outreach</a> ***<!--makes the words take the user to the defind href="" location when clicked-->***

<a href="Competitions.html">Game Manual</a***> <!--makes the words take the user to the defind href="" location when clicked-->***

</div>

<span style="font-size:45px;cursor:pointer;color:white;font-family: Arial" onclick="openMenu()">&#9776; Menu</span> ***<!--styles the open menu icon including making the cursor a pointer while above the element, sets the font, and calls the function openMenu() to open the menu when clicked. The wierd charcers infornt of menu makes the three bars symbol.-->***

<script> <!--defines when javascript starts-->

function openMenu() { ***/\*function called when the user clicks on the span with the menu in the corner\*/***

document.getElementById("Menu").style.width = "250px"; ***<!--changes the width to make the side bar slide out assisted with the transition in css-->***

document.getElementById("Menu").style.border = "10px solid #b3b3ff"; ***<!--makes the border around the element come with the sidebar-->***

}

function closeMenu() {

document.getElementById("Menu").style.width = "0"; ***<!--changes the width to make the side bar slide in assisted with the transition in css-->***

document.getElementById("Menu").style.border = "0px"; ***<!--makes it so the border goes away when the side bar goes away-->***

}

</script> ***<!--defines when javascript ends-->***

<div class= "Main\_Content"> ***<!-- container for the actual content of the page-->***

<div class = "header"> ***<!--containes the header mainly in order to make the background-->***

<img class="content1" src="Team/TEAM.svg"> ***<!--the actual image inserted as the header title-->***

</div>

<br>***<!--puts a break effectivly pushes things down-->***

<img class= "Main\_Image" src="SliceLife.gif"> ***<!--inserts the main image displayed-->***

<div class = "buttons" style="width:100%">***<!--Defines the container for the*** ***buttons-->***

<button style="width:19.74%" onclick="window.location.href = 'About.html';">About</button> ***<!- for all of theses it defines what the buttons are called in css the width src="" is where the button takes them and the words in between the opener and closer are the words displayed on the button-->***

<button style="width:19.74%" onclick="window.location.href = 'Subteams.html';">Sub-teams</button>

<button style="width:19.74%" onclick="window.location.href = 'Accomplishments.html';">Accomplishments</button>

<button style="width:19.74%" onclick="window.location.href = 'Outreach.html';">Outreach</button>

</div>

</div>

</body>

</html>

## About

### Overview

Content: Contains some basic information about blue cheese robotics in a presentation format.

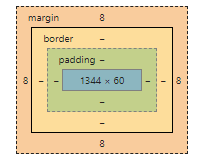
Purpose: To give a basic look at the team.

Audience: Non-blue cheese affiliated users.

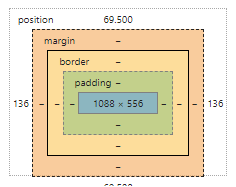
### Layout



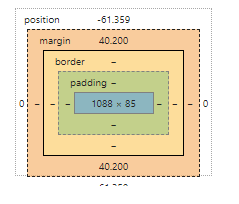
1. body



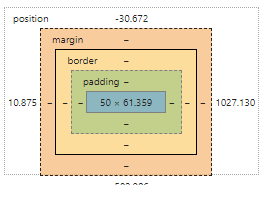
1. Main\_Content



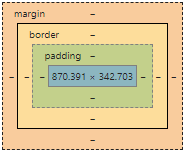
1. title



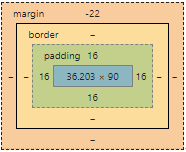
1. back



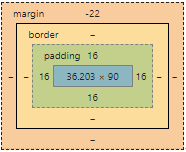
1. mySlides fade



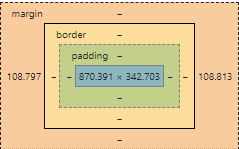
1. prev



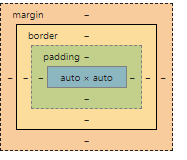
1. next



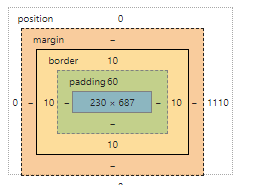
1. slideshow-container



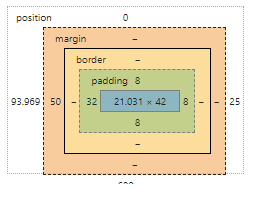
1. menu



1. Menu



1. close



### Code

<!DOCTYPE html>

<html lang="en"> ***<!--set up that this is an html file and the language is english-->***

<head>

<meta charset="UTF-8"> ***<!--defines the type of charcters being used-->***

<style>

***/\*sets settings and background\*/***

\* { ***/\*all sellector\*/***

box-sizing: border-box; ***/\*The CSS box-sizing property allows us to include the padding and border in an element’s total width and height. The border box variable makes it so element padding and border are included in the width and height\*/***

}

body { ***/\*effectivly all selector\*/***

background-image: url("bgtile.jpg"); ***/\*sets the background image\*/***

}

***/\*CSS for the main content area\*/***

.Main\_Content { ***/\*selector for main content area\*/***

position: absolute; ***/\*The position property specifies the type of positioning method used for an element. The absolute variable means The element is positioned relative to anything it is containted within. In this case the body. \*/***

width:80%; ***/\*sets the width to 80%\*/***

height:80%; ***/\*sets the height to 80%\*/***

right:10%; ***/\*puts it 10% from the right\*/***

top:10%; ***/\*sets it 10% from the top\*/***

background-color: white;

}

***/\* CSS for the Central Image \*/***

.Main\_Image { ***/\*selector for image\*/***

display: block; ***/\*The display property specifies the display behavior (the type of rendering box) of an element. The block variable means it displays*** an element as a block element (like <p>). It starts on a new line, and takes up the whole width\*/

margin-left: auto***;/\*effectly this in combination with the dispaly setting centers an element\*/***

margin-right: auto;

width: 40%; ***/\*sets width to 40%\*/***

}

***/\*CSS for the side menu bar\*/***

***/\* Set up the side bar menu itself. Javascript will make it so that it will appear by chaning the width from zero \*/***

.Menu { ***/\*selects the overall menu section\*/***

height: 100%; ***/\*sets the height to 100%\*/***

width: 0; ***/\*changed in javascript when you click to open the menu\*/***

position: fixed; ***/\*The position property specifies the type of positioning method used for an element. The fixed variable means that the element is positioned relative to the browser window\*/***

z-index: 4; ***/\*The z-index property specifies the stack order of an element. An element with greater stack order is always in front of an element with a lower stack order.\*/***

top: 0; ***/\*puts it at the top\*/***

left: 0***; /\*puts it to the left\*/***

background-color: #fafafa;

overflow-x: hidden; ***/\*The overflow-x property specifies whether to clip the content, add a scroll bar, or display overflow content of a block-level element, when it overflows at the left and right edges. The hidden variable makes it so the content is clipped - and no scrolling mechanism is provided\*/***

transition: 0.2s; ***/\*Specifies how many seconds or milliseconds the transition effect takes to complete. In this case the transition form 0 width to full side bar width\*/***

padding-top: 60px; : ***/\*Padding creates extra space within an element on the top\*/***

}

***/\*Design for the words within the side bar menu\*/***

.Menu a { ***/\*selects the words within the side bar menu\*/***

padding: 8px 8px 8px 32px; ***/\* sets extra space within the element in the order of top, right, bottom, then left\*/***

text-decoration: none; ***/\*takes away the standard underline in a link\*/***

font-size: 25px;

font-family: Arial***;/\*sets the font\*/***

color: black;

display: block; ***/\*The display property specifies the display behavior (the type of rendering box) of an element. The block variable means it displays an element as a block element (like <p>). It starts on a new line, and takes up the whole width\*/***

transition: 0.1s; ***/\*Specifies how many seconds or milliseconds the transition effect takes to complete. In this case the transition form 0 width to full side bar width\*/***

}

***/\*Positions and designs the menu close button\*/***

.Menu .close { ***/\*selects the close icon within the menu sidebar\*/***

position: absolute; ***/\*The position property specifies the type of positioning method used for an element. The absolute variable means The element is positioned relative to anything it is containted within. In this case the sidebar menu. \*/***

top: 0; ***/\*puts it at the top\*/***

right: 25px; ***/\*puts it that amount from the right\*/***

font-size: 36px;

margin-left: 50px; ***/\*margin creates extra space around an element in this case to the left\*/***

}

.content1 { ***/\*selects the first content section under the image\*/***

text-align: center; ***/\*The text-align property specifies the horizontal alignment of text in an element. In this case it is to the center\*/***

position:relative; ***/\*The position property specifies the type of positioning method used for an element. The variable realtive means the element is positioned relative to its normal position, so "left:20px" adds 20 pixels to the element's LEFT position\*/***

}

#title{ ***/\*selects the header\*/***

text-align: center; ***/\*The text-align property specifies the horizontal alignment of text in an element. In this case it is to the center\*/***

font-size: 60px;

color: white;

background-color:#000066;

position:relative; ***/\*The position property specifies the type of positioning method used for an element. The variable realtive means the element is positioned relative to its normal position, so "left:20px" adds 20 pixels to the element's LEFT position\*/***

top:-10%; /***\* it is moved up relative to where it used to be\*/***

font-family: Arial Black; ***/\*sets the font\*/***

z-index: 1; ***/\*The z-index property specifies the stack order of an element. An element with greater stack order is always in front of an element with a lower stack order.\*/***

}

.back{ ***/\*selects the character of the back button\*/***

z-index: 3; ***/\*The z-index property specifies the stack order of an element. An element with greater stack order is always in front of an element with a lower stack order.\*/***

position: absolute; ***/\*The position property specifies the type of positioning method used for an element. The absolute variable means The element is positioned relative to anything it is containted within. In this case the body. \*/***

top:-5%; ***/\* it is moved up relative to where it used to be\*/***

color: white;

width:50px***;/\*sets the width to 50 pixels\*/***

height:10%; ***/\*sets the height to 10%\*/***

left:1%; ***/\*it is moved a little form the left\*/***

font-family: Arial; ***/\*sets the font\*/***

}

.backBG{ ***/\*selects the baclground of the back button\*/***

height: 85px; ***/\*it has a height of 85 pixels\*/***

width: 85px; ***/\*it has a width of 85 pixels\*/***

background-color: #4d6e8f;

z-index: 2; ***/\*The z-index property specifies the stack order of an element. An element with greater stack order is always in front of an element with a lower stack order.\*/***

position: absolute; ***/\*The position property specifies the type of positioning method used for an element. The absolute variable means The element is positioned relative to anything it is containted within. In this case the sidebar menu. \*/***

top:-3.4%; ***/\* it is moved up relative to where it used to be\*/***

left:0%; ***/\*it is all the way to the left\*/***

opacity: 0.8***;/\*makes it semi-tranparent\*/***

}

.mySlides {display: none} ***/\*hides the slides until it is their turn\*/***

img {vertical-align: middle;} ***/\*puts all the img in the middle from a vertical perspective\*/***

***/\* Slideshow container \*/***

.slideshow-container { ***/\*selector for what holds the presentation\*/***

width: 80%; ***/\*sets width to 80%\*/***

position: relative; ***/\*The position property specifies the type of positioning method used for an element. The variable realtive means the element is positioned relative to its normal position, so "left:20px" adds 20 pixels to the element's LEFT position\*/***

right:-12%; ***/\*moves it less the right from where it is automatically positioned\*/***

margin: auto;/***\*centers it\*/***

}

***/\* Next & previous buttons \*/***

.prev, .next { /***\*selects for the css the next and previous button\*/***

cursor: pointer; ***/\*sets the cursor to the pointer hand while cursor over it\*/***

position: absolute; ***/\*The position property specifies the type of positioning*** ***method used for an element. The absolute variable means The element is positioned relative to anything it is containted within. In this case the body. \*/***

top: 50%; ***/\*it ismoved 50% from the top\*/***

width: auto; /***\*the width isauto done base don what it needs base don font settings\*/***

padding: 16px; ***/\*Padding creates extra space within an element\*/***

margin-top: -22px; ***/\*moves it up from where it usually is\*/***

color: white;

font-weight: bold; ***/\*just like in word it bolds the text\*/***

font-size: 68px;

transition: 0.6s ease; ***/\*sets transition duration and setting\*/***

border-radius: 0 3px 3px 0; ***/\*sets the roundness of the borders on the top, right, bottom, then left\*/***

color:blue;

user-select: none;

}

***/\* Position the "next button" to the right \*/***

.next {

right: 15%; /***\*moves it 15% from the right\*/***

border-radius: 3px 0 0 3px; ***/\*sets the roundness of the borders on the top, right, bottom, then left\*/***

}

.prev {

left: 15%; ***/moves it 15% from the left\*/***

}

***/\* Fading animation \*/***

.fade {

-webkit-animation-name: fade; ***/\*says what type of animation is happening\*/***

-webkit-animation-duration: 1.5s; ***/\*sets the duration for the animation \*/***

animation-name: fade; ***/\*says what type of animation is happening\*/***

animation-duration: 1.5s; ***/\*sets the duration for the animation \*/***

}

@-webkit-keyframes fade {/\*sets the settings for the animation duration\*/

from {opacity: .4} ***/\*it goes from opacity .4\*/***

to {opacity: 1} ***/\*to opacity 1\*/***

}

@keyframes fade { /sets the settings for the animation duration\*/

from {opacity: .4} ***/\*it goes from opacity .4\*/***

to {opacity: 1} ***/\*to opacity 1\*/***

}

</style>

</head>

<body>

<div id="Menu" class="Menu"> ***<!--sets up the container for the side bar menu-->***

<a href="javascript:void(0)" class="close" onclick="closeMenu()">&times;</a> ***<!-created the close icon that on click actiacted the closeMenu() function. The symbol is created using a char code-->***

<a href="FIRST.html">FIRST</a> ***<!--makes the words take the user to the defind href="" location when clicked-->***

<a href="Team.html">Blue Cheese Robotics</a> ***<!--makes the words take the user to the defind href="" location when clicked-->***

<a href="Compeitions.html">Competitions</a> ***<!--makes the words take the user to the defind href="" location when clicked-->***

<a href="Outreach.html">Our Outreach</a> ***<!--makes the words take the user to the defind href="" location when clicked-->***

</div>

<span style="font-size:40px;cursor:pointer;color:white;font-family: Arial;" onclick="openMenu()">&#9776; Menu</span> ***<!--styles the open menu icon including making the cursor a pointer while above the element, sets the font, and calls the function openMenu() to open the menu when clicked. The wierd charcers infornt of menu makes the three bars symbol.-->***

<script> ***<!--defines when javascript starts-->***

function openMenu() { ***//function called when the user clicks on the span with the menu in the corner***

document.getElementById("Menu").style.width = "250px"; ***//changes the width to make the side bar slide out assisted with the transition in css***

document.getElementById("Menu").style.border = "10px solid #b3b3ff"; ***//makes the border around the element come with the sidebar***

}

function closeMenu() {

document.getElementById("Menu").style.width = "0"; ***<!--changes the width to make the side bar slide in assisted with the transition in css-->***

document.getElementById("Menu").style.border = "0px"; ***<!--makes it so the border goes away when the side bar goes away-->***

}

</script> ***<!--defines when javascript ends-->***

<div class= "Main\_Content"> ***<!-- container for the actual content of the page-->***

<a href="Team.html"> ***<!--makes everythign inside a link to href=""-->***

<span class="back"style='font-size:85px;'>&#8249;&#8249;</span> <***!-=creates the back button-->***

<span class="backBG"></span>

</a>

<h1 id="title">About</h1>***<!--creates the header at the top-->***

<div class="slideshow-container"> ***<!--the container that holds the slideshow-->***

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="about\_slides/Slide1.png" style="width:70%">  ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="about\_slides/Slide2.png" style="width:70***%"> <!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="about\_slides/Slide3.png" style="width:70***%"> <!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="about\_slides/Slide4.png" style="width:70***%"> <!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="about\_slides/Slide5.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

</div>

<a class="prev" onclick="plusSlides(-1)">&#10094;</a> ***<!--link that causes the slides to go backwards-->***

<a class="next" onclick="plusSlides(1)">&#10095;</a> ***<!--link that causes the slides to go forwards-->***

</div>

<script>

var slideIndex = 1***;//creates an index to be used in the function that at this moment is one***

showSlides(slideIndex); ***//calls the function putting in the set slideindex of 1***

function plusSlides(n) { ***//when you want to go to the next slide***

showSlides(slideIndex += n); ***//this calls the show slides function giving it one number farther to get the slide you want***

}

function currentSlide(n) { ***// when you want to go to the previous slide***

showSlides(slideIndex = n); ***//you go to one below this***

}

function showSlides(n) {***//function called in the beggining with n being an inputted variable uses recursive functions to find where we are supposed to be***

var i; ***//creates a variable i***

var slides = document.getElementsByClassName("mySlides"); ***//creates a variable slides that equal our slides we created in html***

if (n > slides.length) {slideIndex = 1***}// makes it so if the counter n is biiger than the amount of slides it goes back to one***

if (n < 1) {slideIndex = slides.length***}//makes it so if you backwards enough to go past the first you go to the last one***

for (i = 0; i < slides.length; i++) {***//goes through all the slides***

slides[i].style.display = "none"; ***//sets each slide to not display***

}

slides[slideIndex-1].style.display = "block"; ***//sets the slide we want to display***

dots[slideIndex-1].className += " active";***// marks the slide as active, so if we*** wanted to do more css in the future we could

}

</script>

</body>

</html>

## Sub teams

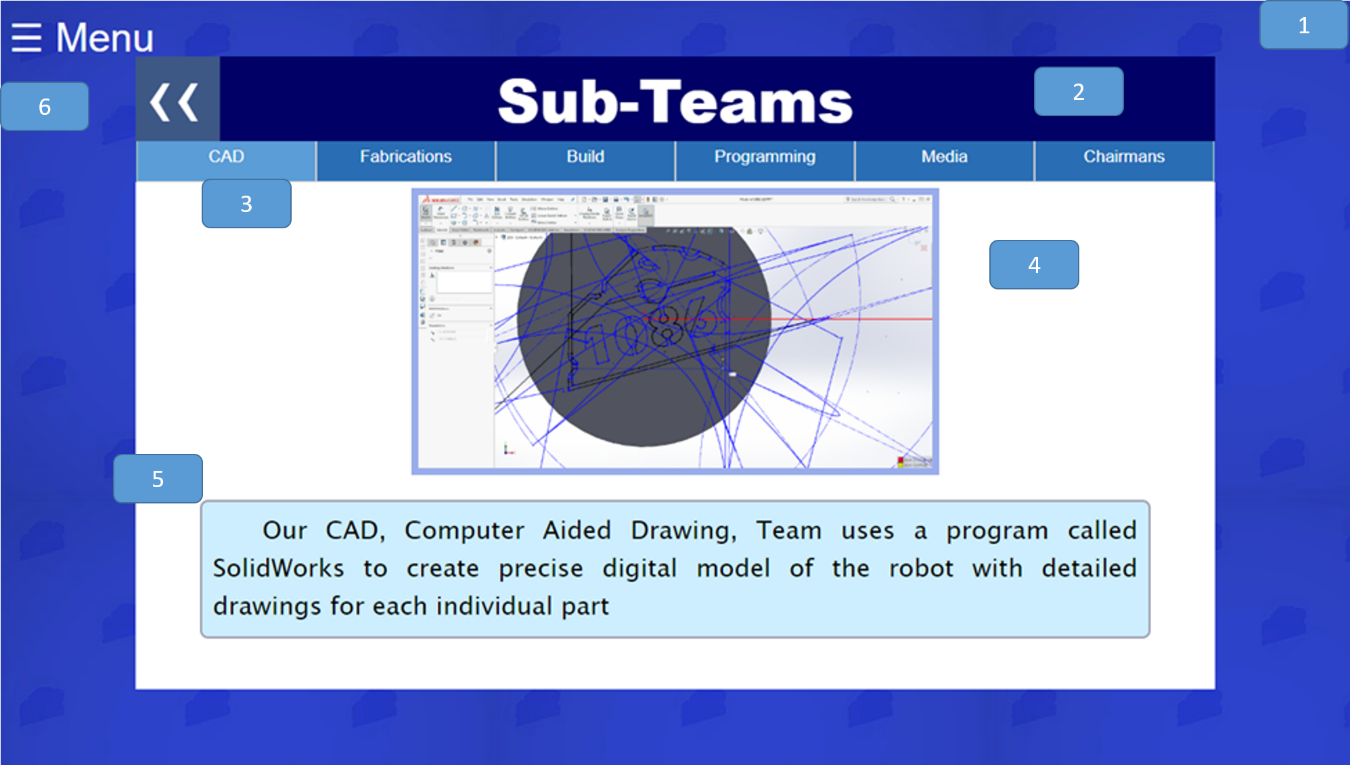
### Overview

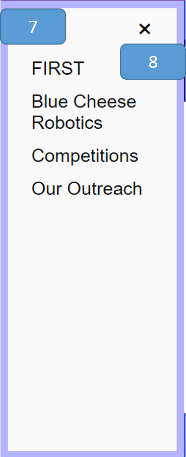
Content: Contains and interactive display of information of the different sub teams with tabs that each contain content about each team.

Purpose: To familiarize users with the structure of the team and our capabilities.

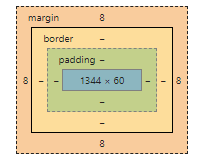
Audience: Non-blue cheese affiliated users.

### Layout

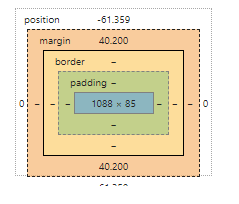




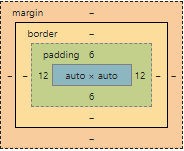
1. body



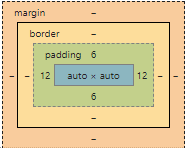
1. title



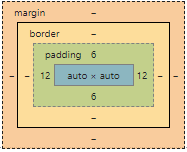
1. tabcontent
   1. CAD



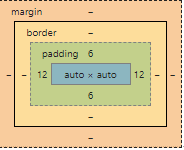
* 1. Fabrications



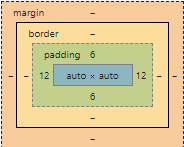
* 1. Build



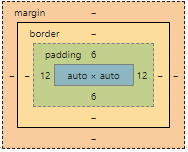
* 1. Programming



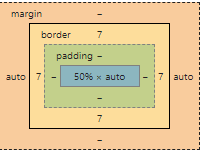
* 1. Media



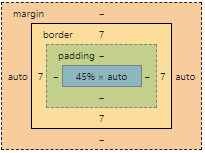
* 1. Chairmans



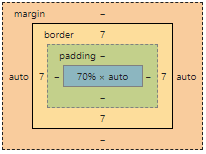
1. Main\_Image
   1. CAD



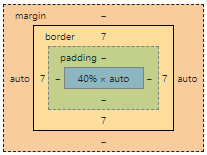
* 1. Fabrications



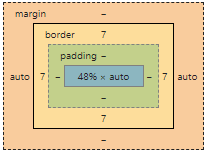
* 1. Build



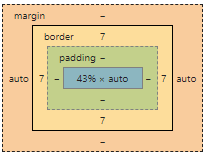
* 1. Programming



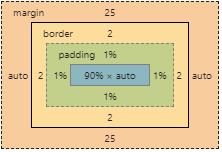
* 1. Media



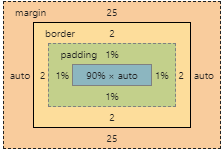
* 1. Chairmans



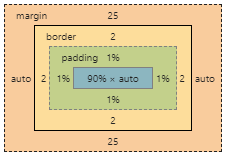
1. p
   1. CAD



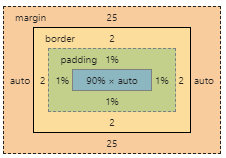
* 1. Fabrications



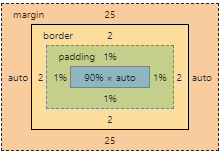
* 1. Build



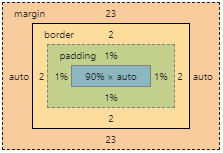
* 1. Programming



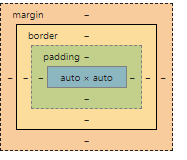
* 1. Media



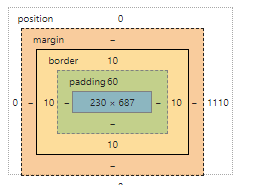
* 1. Chairmans



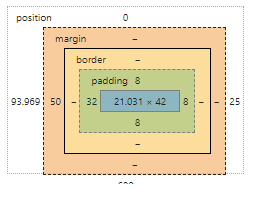
1. menu



1. Menu



1. close



### Code

<!DOCTYPE html>

<html lang="en"> ***<!--set up that this is an html file and the language is english-->***

<head>

<meta charset="UTF-8"> ***<!--defines the type of charcters being used-->***

<style>

***/\*sets settings and background\*/***

\* { ***/\*all sellector\*/***

box-sizing: border-box; ***/\*The CSS box-sizing property allows us to include the padding and border in an element’s total width and height. The border box variable makes it so element padding and border are included in the width and height\*/***

}

body { ***/\*effectivly all selector\*/***

background-image: url("bgtile.jpg"); ***/\*sets the background image\*/***

}

***/\*CSS for the main content area\*/***

.Main\_Content { ***/\*selector for main content area\*/***

position: absolute; ***/\*The position property specifies the type of positioning method used for an element. The absolute variable means The element is positioned relative to anything it is containted within. In this case the body. \*/***

width:80%; ***/\*sets the width to 80%\*/***

height:80%; ***/\*sets the height to 80%\*/***

right:10%; ***/\*puts it 10% from the right\*/***

top:10%; ***/\*sets it 10% from the top\*/***

background-color: white;

}

.Main\_Image {

display: block;

margin-left: auto;

margin-right: auto;

border: 7px solid rgba(0,51,202,.4);

}

***/\*CSS for the side menu bar\*/***

***/\* Set up the side bar menu itself. Javascript will make it so that it will appear by chaning the width from zero \*/***

.Menu { ***/\*selects the overall menu section\*/***

height: 100%; ***/\*sets the height to 100%\*/***

width: 0; ***/\*changed in javascript when you click to open the menu\*/***

position: fixed; ***/\*The position property specifies the type of positioning method used for an element. The fixed variable means that the element is positioned relative to the browser window\*/***

z-index: 4; ***/\*The z-index property specifies the stack order of an element. An element with greater stack order is always in front of an element with a lower stack order.\*/***

top: 0; ***/\*puts it at the top\*/***

left: 0; ***/\*puts it to the left\*/***

background-color: #fafafa;

overflow-x: hidden; ***/\*The overflow-x property specifies whether to clip the content, add a scroll bar, or display overflow content of a block-level element, when it overflows at the left and right edges. The hidden variable makes*** it so the content is clipped - and no scrolling mechanism is provided\*/

transition: 0.2s; ***/\*Specifies how many seconds or milliseconds the transition effect takes to complete. In this case the transition form 0 width to full side bar width\*/***

padding-top: 60px; : ***/\*Padding creates extra space within an element on the top\*/***

}

***/\*Design for the words within the side bar menu\*/***

.Menu a { ***/\*selects the words within the side bar menu\*/***

padding: 8px 8px 8px 32px; ***/\* sets extra space within the element in the order of top, right, bottom, then left\*/***

text-decoration: none; ***/\*takes away the standard underline in a link\*/***

font-size: 25px;

font-family: Arial;***/\*sets the font\*/***

color: black;

display: block; ***/\*The display property specifies the display behavior (the type of rendering box) of an element. The block variable means it displays an element as a block element (like <p>). It starts on a new line, and takes up the whole width\*/***

transition: 0.1s; ***/\*Specifies how many seconds or milliseconds the transition effect takes to complete. In this case the transition form 0 width to full side bar width\*/***

}

***/\*Positions and designs the menu close button\*/***

.Menu .close { ***/\*selects the close icon within the menu sidebar\*/***

position: absolute; ***/\*The position property specifies the type of positioning method used for an element. The absolute variable means The element is positioned relative to anything it is containted within. In this case the sidebar menu. \*/***

top: 0; ***/\*puts it at the top\*/***

right: 25px; ***/\*puts it that amount from the right\*/***

font-size: 36px;

margin-left: 50px; ***/\*margin creates extra space around an element in this case to the left\*/***

}

.content1 {***/\*selects the first content section under the image\*/***

text-align: center;***/\*The text-align property specifies the horizontal alignment of text in an element. In this case it is to the center\*/***

}

.tab { ***/\*selects the buttons appreaing at the top of the page\*/***

overflow: hidden; ***/\*The overflow-x property specifies whether to clip the content, add a scroll bar, or display overflow content of a block-level element, when it overflows at the left and right edges. The hidden variable makes it so the content is clipped - and no scrolling mechanism is provided\*/***

border: 1px solid #296eb4; ***/\*this creates a border around the element it is define by width, what type, then color\*/***

border-style: none solid none solid; ***/\*this defined the border on each side in the order of top, right, bottom, then left\*/***

background-color: #296eb4;

position:relative***; /\*The position property specifies the type of positioning method used for an element. The variable realtive means the element is positioned relative to its normal position, so "left:20px" adds 20 pixels to the element's LEFT position\*/***

top:-17.2%; ***/\*makes it higher than it would usually be\*/***

}

.tab button { ***/\*selects the button part that acts as a button of the tabs at the top of the page\*/***

background-color: inherit***;/\*same background color as tab as it is it's parent element or elmenet that it is within\*/***

float: left;

cursor: pointer; ***/\*sets the cursor to the pointer hand while cursor over it\*/***

padding: 14px 16px;

transition: 0.3s;

font-size: 17px;

border: 1px solid #ccc; ***/\*this creates a border around the element it is define by width, what type, then color\*/***

width:16.666666666666666666666666666667%;

color: white;

font-family: "Lucida Sans Unicode"; ***/\*sets the font\*/***

}

.tab button.active { ***/\*choose the tab that is in use \*/***

background-color: #5b9dd7;

}

.tabcontent { ***/\*selects the content displayed after clicking the buttons\*/***

display: none; ***/\*Hides content until javascript changes\*/***

padding: 6px 12px;

text-indent: 50px***; /\*The text-indent property specifies the indentation of the first line in a text-block.\*/***

text-align: justify; ***/\*The text-align property specifies the horizontal alignment of text in an element. The jusify variable means it stretches the lines so that each line has equal width (like in newspapers and magazines)\*/***

position:relative; ***/\*The position property specifies the type of positioning method used for an element. The variable realtive means the element is positioned relative to its normal position, so "left:20px" adds 20 pixels to the element's LEFT position\*/***

top:-18%;***/\* it is moved up relative to where it used to be\*/***

}

#title{ ***/\*selects the header\*/***

text-align: center; ***/\*The text-align property specifies the horizontal alignment of text in an element. In this case it is to the center\*/***

font-size: 60px;

color: white;

background-color:#000066;

position:relative; ***/\*The position property specifies the type of positioning method used for an element. The variable realtive means the element is positioned relative to its normal position, so "left:20px" adds 20 pixels to the element's LEFT position\*/***

top:-10%; ***/\* it is moved up relative to where it used to be\*/***

font-family: Arial Black***; /\*sets the font\*/***

z-index: 1; ***/\*The z-index property specifies the stack order of an element. An element with greater stack order is always in front of an element with a lower stack order.\*/***

}

p{

text-indent: 50px; ***/\*The text-indent property specifies the indentation of the first line in a text-block.\*/***

text-align: justify; ***/\*The text-align property specifies the horizontal alignment of text in an element. The jusify variable means it stretches the lines so that each line has equal width (like in newspapers and magazines)\*/***

border: 2px solid #9d9daf; ***/\*this creates a border around the element it is define by width, what type, then color\*/***

border-radius: 8px; /***\*this rounds the border around an element\*/***

padding: 1%; ***/\*Padding creates extra space within an element\*/***

display: block; ***/\*The display property specifies the display behavior (the type of rendering box) of an element. The block variable means it displays an element as a block element (like <p>). It starts on a new line, and takes up the whole width\*/***

margin-left: auto; ***/\*effectly this in combination with the dispaly setting centers an element\*/***

margin-right: auto;

font-size:25px;

font-family: "Lucida Sans Unicode"; ***/\*sets the font\*/***

background-color: #cceeff ***/\*creates the background color\*/***

}

.back{ ***/\*selects the character of the back button\*/***

z-index: 3; ***/\*The z-index property specifies the stack order of an element. An element with greater stack order is always in front of an element with a lower stack order.\*/***

position: absolute; ***/\*The position property specifies the type of positioning method used for an element. The absolute variable means The element is positioned relative to anything it is containted within. In this case the body. \*/***

top:-5%; ***/\* it is moved up relative to where it used to be\*/***

color: white;

width:50px;***/\*sets the width to 50 pixels\*/***

height:10%; ***/\*sets the height to 10%\*/***

left:1%; ***/\*it is moved a little form the left\*/***

font-family: Arial; ***/\*sets the font\*/***

}

.backBG{ ***/\*selects the baclground of the back button\*/***

height: 85px; ***/\*it has a height of 85 pixels\*/***

width: 85px; ***/\*it has a width of 85 pixels\*/***

background-color: #4d6e8f;

z-index: 2; ***/\*The z-index property specifies the stack order of an element. An element with greater stack order is always in front of an element with a lower stack order.\*/***

position: absolute; ***/\*The position property specifies the type of positioning method used for an element. The absolute variable means The element is positioned relative to anything it is containted within. In this case the sidebar menu. \*/***

top:-3.4%; ***/\* it is moved up relative to where it used to be\*/***

left:0%; ***/\*it is all the way to the left\*/***

opacity: 0.8;***/\*makes it semi-tranparent\*/***

}

</style>

</head>

<body>

<div id="Menu" class="Menu"> ***<!--sets up the container for the side bar menu-->***

<a href="javascript:void(0)" class="close" onclick="closeMenu()">&times;</a> ***<!-created the close icon that on click actiacted the closeMenu() function. The symbol is created using a char code-->***

<a href="FIRST.html">FIRST</a> ***<!--makes the words take the user to the defind href="" location when clicked-->***

<a href="Team.html">Blue Cheese Robotics</a***> <!--makes the words take the user to the defind href="" location when clicked-->***

<a href="Compeitions.html">Competitions</a> ***<!--makes the words take the user to the defind href="" location when clicked-->***

<a href="Outreach.html">Our Outreach</a> ***<!--makes the words take the user to the defind href="" location when clicked-->***

</div>

<span style="font-size:40px;cursor:pointer;color:white;font-family: Arial;" onclick="openMenu()">&#9776; Menu</span> ***<!--styles the open menu icon including making the cursor a pointer while above the element, sets the font, and calls the function openMenu() to open the menu when clicked. The wierd charcers infornt of menu makes the three bars symbol.-->***

<script> <!--defines when javascript starts-->

function openMenu() { ***//function called when the user clicks on the span with the menu in the corner***

document.getElementById("Menu").style.width = "250px"; ***//changes the width to make the side bar slide out assisted with the transition in css***

document.getElementById("Menu").style.border = "10px solid #b3b3ff"; ***//makes the border around the element come with the sidebar***

}

function closeMenu() {

document.getElementById("Menu").style.width = "0"; ***<!--changes the width to make the side bar slide in assisted with the transition in css-->***

document.getElementById("Menu").style.border = "0px"; ***<!--makes it so the border goes away when the side bar goes away-->***

}

</script> ***<!--defines when javascript ends-->***

<div class= "Main\_Content"> ***<!-- container for the actual content of the page-->***

<a href="Team.html"> ***<!--makes everythign inside a link to href=""-->***

<span class="back"style='font-size:85px;'>&#8249;&#8249;</span> ***<!-=creates the back button-->***

<span class="backBG"></span>

</a>

<h1 id="title">Subteams</h1><***!--creates the header at the top-->***

<div class="tab"> ***<!--creates the tabs dsiplayed at the top-->***

<button class="tablinks" onclick="openNow(event, 'CAD')">CAD</button> ***<!--creates the individidual link for each tab and has it call a function giving the function what it needs to display the right thing-->***

<button class="tablinks" onclick="openNow(event, 'Fabrications')">Fabrications</button>

<button class="tablinks" onclick="openNow(event, 'Build')">Build</button>

<button class="tablinks" onclick="openNow(event, 'Programming')">Programming</button>

<button class="tablinks" onclick="openNow(event, 'Media')">Media</button>

<button class="tablinks" onclick="openNow(event, 'Chairmans')">Chairmans</button>

</div>

<div id="CAD" class="tabcontent"> <!--holds the tab content-->

<img class= "Main\_Image" src="Subteams/CAD.png" style="width:50%;"> ***<!-- the img to represent the subteam-->***

<p>Our CAD, Computer Aided Drawing, Team uses a program called SolidWorks to create precise digital model of the robot with detailed drawings for each individual part</p> ***<!--holds the descriptor about the subteam-->***

</div>

<div id="Fabrications" class="tabcontent"> <!--holds the tab content-->

<img class= "Main\_Image" src="Subteams/Fabrications.JPG" top="10%" style="width:45%;"> ***<!-- the img to represent the subteam-->***

<p>Our Fabrications Team using tools like: band saw, drill press, and CNC mill to create the majority of our parts for the robot.</p> ***<!--holds the descriptor about the subteam-->***

</div>

<div id="Build" class="tabcontent"> ***<!--holds the tab content-->***

<img class= "Main\_Image" src="Subteams/build.png" top="10%" style="width:70%;"> ***<!-- the img to represent the subteam-->***

<p>Our Build Team makes the CAD or digital diagram of the robot a reality by putting together all the pieces.</p> ***<!--holds the descriptor about the subteam-->***

</div>

<div id="Programming" class="tabcontent"> ***<!--holds the tab content-->***

<img class= "Main\_Image" src="Subteams/Programming.JPG" top="10%" style="width:40%;"> ***<!-- the img to represent the subteam-->***

<p>Our Programming Team uses Java and Git Hub to program the robot and Android Java and XML to create an app to record data during competitions.</p> ***<!--holds the descriptor about the subteam-->***

</div>

<div id="Media" class="tabcontent"> ***<!--holds the tab content-->***

<img class= "Main\_Image" src="Subteams/media.JPG" top="10%" style="width:48%;"> ***<!-- the img to represent the subteam-->***

<p>Our Media Team creatse videos, photos, logo designs, shirt designs, animation, and any other digital media.</p> ***<!--holds the descriptor about the subteam-->***

</div>

<div id="Chairmans" class="tabcontent"> ***<!--holds the tab content-->***

<img class= "Main\_Image" src="Subteams/Chairmans.JPG" top="10%" style="width:43%;"> ***<!-- the img to represent the subteam-->***

<p style="font-size:23px;">Our Chairmans team write essay and give presentations on what makes our team special and what we do to spread STEM throughout our community and the world.</p> ***<!--holds the descriptor about the subteam-->***

</div>

<script>

function openNow(evt, dtls) { ***//is function callee when the click on a button that takes in information about which buttton it is in order to show the right thing***

var i, tabcontent, tablinks; ***//creates or initializes these variables***

tabcontent = document.getElementsByClassName("tabcontent***");//sets the variable tabcontnet to the tabcontent we have already made in html***

for (i = 0; i < tabcontent.length; i++) { ***//goes through and make the display none again for all***

tabcontent[i].style.display = "none";

}

tablinks = document.getElementsByClassName("tablinks"); ***//get the tablinks in the tab and makes them active if they are not already in order to chnage the color***

for (i = 0; i < tablinks.length; i++) {

tablinks[i].className = tablinks[i].className.replace(" active", "");

}

document.getElementById(dtls).style.display = "block"; ***//makes the one has been clicked on dispaly.***

evt.currentTarget.className += " active"***;//marks the only one supposed to be active as active***

}

</script>

</body>

</html>

## Accomplishments

### Overview

Content: Contains information of the major accomplishments of the team in presentation format.

Purpose: In order to convey the general idea that the team is successful

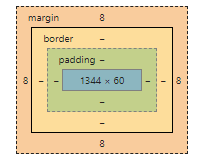
Audience: Non-blue cheese affiliated users.

### Layout

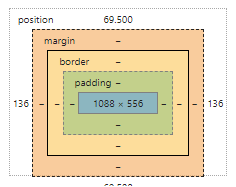




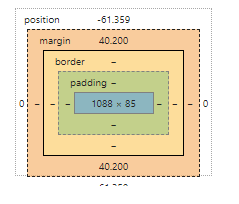
1. body



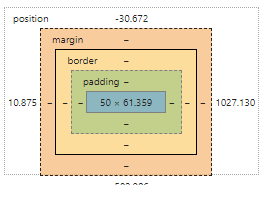
1. Main\_Content



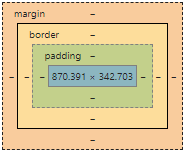
1. title



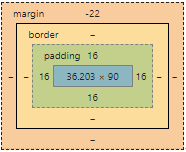
1. back



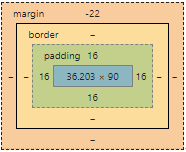
1. mySlides fade



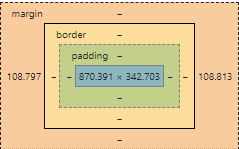
1. prev



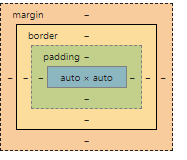
1. next



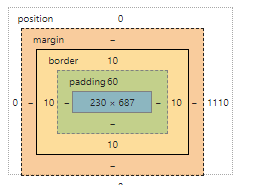
1. slideshow-container



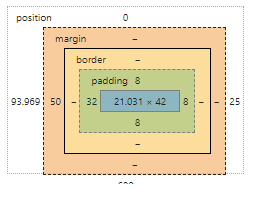
1. menu



1. Menu



1. close



### Code

<!DOCTYPE html>

<html lang="en"> ***<!--set up that this is an html file and the language is english-->***

<head>

<meta charset="UTF-8"> ***<!--defines the type of charcters being used-->***

<style>

***/\*sets settings and background\*/***

\* { ***/\*all sellector\*/***

box-sizing: border-box; ***/\*The CSS box-sizing property allows us to include the padding and border in an element’s total width and height. The border box variable makes it so element padding and border are included in the width and height\*/***

}

body { ***/\*effectivly all selector\*/***

background-image: url("bgtile.jpg"); ***/\*sets the background image\*/***

}

***/\*CSS for the main content area\*/***

.Main\_Content { ***/\*selector for main content area\*/***

position: absolute; ***/\*The position property specifies the type of positioning method used for an element. The absolute variable means The element is positioned relative to anything it is containted within. In this case the body. \*/***

width:80%; ***/\*sets the width to 80%\*/***

height:80%; ***/\*sets the height to 80%\*/***

right:10%; ***/\*puts it 10% from the right\*/***

top:10%; ***/\*sets it 10% from the top\*/***

background-color: white;

}

***/\* CSS for the Central Image \*/***

.Main\_Image { ***/\*selector for image\*/***

display: block; ***/\*The display property specifies the display behavior (the type of rendering box) of an element. The block variable means it displays an element as a block element (like <p>). It starts on a new line, and takes up the whole width\*/***

margin-left: auto***;/\*effectly this in combination with the dispaly setting centers an element\*/***

margin-right: auto;

width: 40%; ***/\*sets width to 40%\*/***

}

***/\*CSS for the side menu bar\*/***

***/\* Set up the side bar menu itself. Javascript will make it so that it will appear by chaning the width from zero \*/***

.Menu { ***/\*selects the overall menu section\*/***

height: 100%; ***/\*sets the height to 100%\*/***

width: 0; ***/\*changed in javascript when you click to open the menu\*/***

position: fixed; ***/\*The position property specifies the type of positioning method used for an element. The fixed variable means that the element is positioned relative to the browser window\*/***

z-index: 4; ***/\*The z-index property specifies the stack order of an element. An element with greater stack order is always in front of an element with a lower stack order.\*/***

top: 0; ***/\*puts it at the top\*/***

left: 0; ***/\*puts it to the left\*/***

background-color: #fafafa;

overflow-x: hidden; ***/\*The overflow-x property specifies whether to clip the content, add a scroll bar, or display overflow content of a block-level element, when it overflows at the left and right edges. The hidden variable makes it so the content is clipped - and no scrolling mechanism is provided\*/***

transition: 0.2s; ***/\*Specifies how many seconds or milliseconds the transition effect takes to complete. In this case the transition form 0 width to full side bar width\*/***

padding-top: 60px; : ***/\*Padding creates extra space within an element on the top\*/***

}

***/\*Design for the words within the side bar menu\*/***

.Menu a { ***/\*selects the words within the side bar menu\*/***

padding: 8px 8px 8px 32px; ***/\* sets extra space within the element in the order of top, right, bottom, then left\*/***

text-decoration: none; ***/\*takes away the standard underline in a link\*/***

font-size: 25px;

font-family: Arial***;/\*sets the font\*/***

color: black;

display: block; ***/\*The display property specifies the display behavior (the type of rendering box) of an element. The block variable means it displays an element as a block element (like <p>). It starts on a new line, and takes up the whole width\*/***

transition: 0.1s; ***/\*Specifies how many seconds or milliseconds the transition effect takes to complete. In this case the transition form 0 width to full side bar width\*/***

}

***/\*Positions and designs the menu close button\*/***

.Menu .close { ***/\*selects the close icon within the menu sidebar\*/***

position: absolute; ***/\*The position property specifies the type of positioning method used for an element. The absolute variable means The element is positioned relative to anything it is containted within. In this case the sidebar menu. \*/***

top: 0; ***/\*puts it at the top\*/***

right: 25px; ***/\*puts it that amount from the right\*/***

font-size: 36px;

margin-left: 50px; ***/\*margin creates extra space around an element in this case to the left\*/***

}

.content1 { ***/\*selects the first content section under the image\*/***

text-align: center; ***/\*The text-align property specifies the horizontal alignment of text in an element. In this case it is to the center\*/***

position:relative; ***/\*The position property specifies the type of positioning method used for an element. The variable realtive means the element is positioned relative to its normal position, so "left:20px" adds 20 pixels to the element's LEFT position\*/***

}

#title{ ***/\*selects the header\*/***

text-align: center; ***/\*The text-align property specifies the horizontal alignment of text in an element. In this case it is to the center\*/***

font-size: 60px;

color: white;

background-color:#000066;

position:relative; ***/\*The position property specifies the type of positioning method used for an element. The variable realtive means the element is positioned relative to its normal position, so "left:20px" adds 20 pixels to the element's LEFT position\*/***

top:-1***0%; /\* it is moved up relative to where it used to be\*/***

font-family: Arial Black; ***/\*sets the font\*/***

z-index: 1; ***/\*The z-index property specifies the stack order of an element. An element with greater stack order is always in front of an element with a lower stack order.\*/***

}

.back{ ***/\*selects the character of the back button\*/***

z-index: 3; ***/\*The z-index property specifies the stack order of an element. An element with greater stack order is always in front of an element with a lower stack order.\*/***

position: absolute; ***/\*The position property specifies the type of positioning method used for an element. The absolute variable means The element is positioned relative to anything it is containted within. In this case the body. \*/***

top:-5%; ***/\* it is moved up relative to where it used to be\*/***

color: white;

width:50px;***/\*sets the width to 50 pixels\*/***

height:10%; ***/\*sets the height to 10%\*/***

left:1%; ***/\*it is moved a little form the left\*/***

font-family: Arial; ***/\*sets the font\*/***

}

.backBG{ ***/\*selects the baclground of the back button\*/***

height: 85px; ***/\*it has a height of 85 pixels\*/***

width: 85px; ***/\*it has a width of 85 pixels\*/***

background-color: #4d6e8f;

z-index: 2; ***/\*The z-index property specifies the stack order of an element. An element with greater stack order is always in front of an element with a lower stack order.\*/***

position: absolute; ***/\*The position property specifies the type of positioning method used for an element. The absolute variable means The element is positioned relative to anything it is containted within. In this case the sidebar menu. \*/***

top:-3.4%; ***/\* it is moved up relative to where it used to be\*/***

left:0%; ***/\*it is all the way to the left\*/***

opacity: 0.8***;/\*makes it semi-tranparent\*/***

}

.mySlides {display: none} ***/\*hides the slides until it is their turn\*/***

img {vertical-align: middle;} ***/\*puts all the img in the middle from a vertical perspective\*/***

***/\* Slideshow container \*/***

.slideshow-container { ***/\*selector for what holds the presentation\*/***

width: 80%; ***/\*sets width to 80%\*/***

position: relative; ***/\*The position property specifies the type of positioning method used for an element. The variable realtive means the element is positioned relative to its normal position, so "left:20px" adds 20 pixels to the element's LEFT position\*/***

right:-12%; ***/\*moves it less the right from where it is automatically positioned\*/***

margin: auto;/***\*centers it\*/***

}

***/\* Next & previous buttons \*/***

.prev, .next { ***/\*selects for the css the next and previous button\*/***

cursor: pointer; ***/\*sets the cursor to the pointer hand while cursor over it\*/***

position: absolute; ***/\*The position property specifies the type of positioning method used for an element. The absolute variable means The element is positioned relative to anything it is containted within. In this case the body. \*/***

top: 50%; ***/\*it ismoved 50% from the top\*/***

width: auto; ***/\*the width isauto done base don what it needs base don font settings\*/***

padding: 16px; ***/\*Padding creates extra space within an element\*/***

margin-top: -22px; ***/\*moves it up from where it usually is\*/***

color: white;

font-weight: bold; ***/\*just like in word it bolds the text\*/***

font-size: 68px;

transition: 0.6s ease; ***/\*sets transition duration and setting\*/***

border-radius: 0 3px 3px 0; ***/\*sets the roundness of the borders on the top, right, bottom, then left\*/***

color:blue;

user-select: none;

}

***/\* Position the "next button" to the right \*/***

.next {

right: 15%; ***/\*moves it 15% from the right\*/***

border-radius: 3px 0 0 3px; ***/\*sets the roundness of the borders on the top, right,*** ***bottom, then left\*/***

}

.prev {

left: 15%; /***moves it 15% from the left\*/***

}

***/\* Fading animation \*/***

.fade {

-webkit-animation-name: fade; ***/\*says what type of animation is happening\*/***

-webkit-animation-duration: 1.5s; ***/\*sets the duration for the animation \*/***

animation-name: fade; ***/\*says what type of animation is happening\*/***

animation-duration: 1.5s; ***/\*sets the duration for the animation \*/***

}

@-webkit-keyframes fade { ***/\*sets the settings for the animation duration\*/***

from {opacity: .4} ***/\*it goes from opacity .4\*/***

to {opacity: 1}***/\*it goes from opacity .4\*/***

}

@keyframes fade { ***/\*sets the settings for the animation duration\*/***

from {opacity: .4} ***/\*it goes from opacity .4\*/***

to {opacity: 1}***/\*it goes from opacity .4\*/***

}

</style>

</head>

<body>

<div id="Menu" class="Menu"> ***<!--sets up the container for the side bar menu-->***

<a href="javascript:void(0)" class="close" onclick="closeMenu()">&times;</a> ***<!-created the close icon that on click actiacted the closeMenu() function. The symbol is created using a char code-->***

<a href="FIRST.html">FIRST</a> ***<!--makes the words take the user to the defind href="" location when clicked-->***

<a href="Team.html">Blue Cheese Robotics</a> ***<!--makes the words take the user to the defind href="" location when clicked-->***

<a href="Compeitions.html">Competitions</a> ***<!--makes the words take the user to the defind href="" location when clicked-->***

<a href="Outreach.html">Our Outreach</a> ***<!--makes the words take the user to the defind href="" location when clicked-->***

</div>

<span style="font-size:40px;cursor:pointer;color:white;font-family: Arial;" onclick="openMenu()">&#9776; Menu</span> ***<!--styles the open menu icon including making the cursor a pointer while above the element, sets the font, and calls the function openMenu() to open the menu when clicked. The wierd charcers infornt of menu makes the three bars symbol.-->***

<script> ***<!--defines when javascript starts-->***

function openMenu() { ***//function called when the user clicks on the span with the menu in the corner***

document.getElementById("Menu").style.width = "250px"; ***//changes the width to make the side bar slide out assisted with the transition in css***

document.getElementById("Menu").style.border = "10px solid #b3b3ff"; ***//makes the border around the element come with the sidebar***

}

function closeMenu() {

document.getElementById("Menu").style.width = "0"; ***<!--changes the width to make the side bar slide in assisted with the transition in css-->***

document.getElementById("Menu").style.border = "0px***"; <!--makes it so the border goes away when the side bar goes away-->***

}

</script> ***<!--defines when javascript ends-->***

<div class= "Main\_Content"> ***<!-- container for the actual content of the page-->***

<a href="Team.html">

<span class="back"style='font-size:85px;'>&#8249;&#8249;</span> ***<!-=creates the back button-->***

<span class="backBG"></span>

</a>

<h1 id="title">Accomplishments</h1> ***<!--creates the header at the top-->***

<div class="slideshow-container"> ***<!--the container that holds the slideshow-->***

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="awards\_slides/Slide2.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade">

<img src="awards\_slides/Slide3.png" style="width:70%">***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade">

<img src="awards\_slides/Slide4.png" style="width:70%">***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade">

<img src="awards\_slides/Slide5.png" style="width:70%">***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade">

<img src="awards\_slides/Slide6.png" style="width:70%">***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade">

<img src="awards\_slides/Slide7.png" style="width:70%">***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade">

<img src="awards\_slides/Slide8.png" style="width:70%">***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

</div>

<a class="prev" onclick="plusSlides(-1)">&#10094;</a***> <!--link that causes the slides to go backwards-->***

<a class="next" onclick="plusSlides(1)">&#10095;</a> ***<!--link that causes the slides to go forwards-->***

</div>

<script>

var slideIndex = 1***;//creates an index to be used in the function that at this moment is one***

showSlides(slideIndex); ***//calls the function putting in the set slideindex of 1***

function plusSlides(n) { ***//when you want to go to the next slide***

showSlides(slideIndex += n***);//this calls the show slides function giving it one number farther to get the slide you want***

}

function currentSlide(n) { ***// when you want to go to the previous slide***

showSlides(slideIndex = n);/***/you go to one below this***

}

function showSlides(n) {***//function called in the beggining with n being an inputted variable uses recursive functions to find where we are supposed to be***

var i; ***//creates a variable i***

var slides = document.getElementsByClassName("mySlides"); ***//creates a variable slides that equal our slides we created in html***

if (n > slides.length) {slideIndex = 1}***// makes it so if the counter n is biiger than the amount of slides it goes back to one***

if (n < 1) {slideIndex = slides.length}***//makes it so if you backwards enough to go past the first you go to the last one***

for (i = 0; i < slides.length; i++) {***//goes through all the slides***

slides[i].style.display = "none"; ***//sets each slide to not display***

}

slides[slideIndex-1].style.display = "block"; ***//sets the slide we want to display***

dots[slideIndex-1].className += " active***";// marks the slide as active, so if we wanted to do more css in the future we could***

}

</script>

</body>

</html>

## Outreach

### Overview

Content: An overview of our outreach projects in a presentation format.

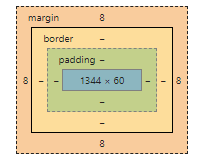
Purpose: To express our outreach efforts and impact in our community and beyond.

Audience: Non-blue cheese affiliated users.

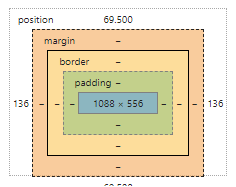
### Layout



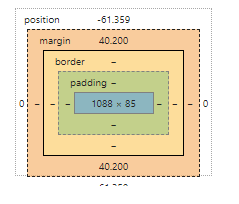
1. body



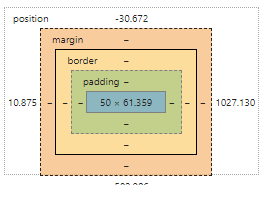
1. Main\_Content



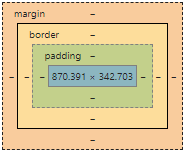
1. title



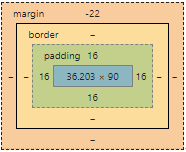
1. back



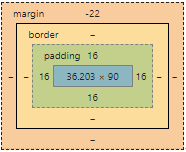
1. mySlides fade



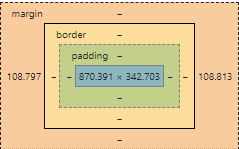
1. prev



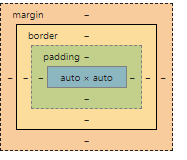
1. next



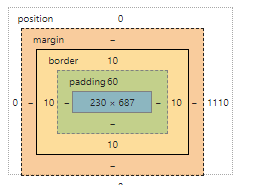
1. slideshow-container



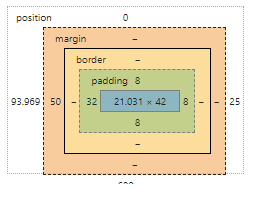
1. menu



1. Menu



1. close



### Code

<!DOCTYPE html>

<html lang="en"> ***<!--set up that this is an html file and the language is english-->***

***<head>***

<meta charset="UTF-8"> ***<!--defines the type of charcters being used-->***

<style>

***/\*sets settings and background***\*/

\* { ***/\*all sellector\*/***

box-sizing: border-box; ***/\*The CSS box-sizing property allows us to include the padding and border in an element’s total width and height. The border box variable makes it so element padding and border are included in the width and height\*/***

}

body { ***/\*effectivly all selector\*/***

background-image: url("bgtile.jpg"); ***/\*sets the background image\*/***

}

***/\*CSS for the main content area\*/***

.Main\_Content { ***/\*selector for main content area\*/***

position: absolute; ***/\*The position property specifies the type of positioning method used for an element. The absolute variable means The element is positioned relative to anything it is containted within. In this case the body. \*/***

width:80%; /***\*sets the width to 80%\*/***

height:80%; ***/\*sets the height to 80%\*/***

right:10%; ***/\*puts it 10% from the right\*/***

top:10%; ***/\*sets it 10% from the top\*/***

background-color: white;

}

***/\* CSS for the Central Image \*/***

.Main\_Image { ***/\*selector for image\*/***

display: block; ***/\*The display property specifies the display behavior (the type of rendering box) of an element. The block variable means it displays an element as a block element (like <p>). It starts on a new line, and takes up the whole width\*/***

margin-left: auto***;/\*effectly this in combination with the dispaly setting centers an element\*/***

margin-right: auto;

width: 30%; ***/\*sets width to 30%\*/***

}

***/\*CSS for the side menu bar\*/***

***/\* Set up the side bar menu itself. Javascript will make it so that it will appear by chaning the width from zero \*/***

.Menu { ***/\*selects the overall menu section\*/***

height: 100%; /***\*sets the height to 100%\*/***

width: 0; ***/\*changed in javascript when you click to open the menu\*/***

position: fixed; ***/\*The position property specifies the type of positioning method used for an element. The fixed variable means that the element is positioned relative to the browser window\*/***

z-index: 4; ***/\*The z-index property specifies the stack order of an element. An element with greater stack order is always in front of an element with a lower stack order.\*/***

top: 0; ***/\*puts it at the top\*/***

left: 0; ***/\*puts it to the left\*/***

background-color: #fafafa;

overflow-x: hidden; ***/\*The overflow-x property specifies whether to clip the content, add a scroll bar, or display overflow content of a block-level element, when it overflows at the left and right edges. The hidden variable makes it so the content is clipped - and no scrolling mechanism is provided\*/***

transition: 0.2s; ***/\*Specifies how many seconds or milliseconds the transition effect takes to complete. In this case the transition form 0 width to full side bar width\*/***

padding-top: 60px; : ***/\*Padding creates extra space within an element on the top\*/***

}

***/\*Design for the words within the side bar menu\*/***

.Menu a { ***/\*selects the words within the side bar menu\*/***

padding: 8px 8px 8px 32px; ***/\* sets extra space within the element in the order of top, right, bottom, then left\*/***

text-decoration: none; ***/\*takes away the standard underline in a link\*/***

font-size: 25px;

font-family: Arial***;/\*sets the font\*/***

color: black;

display: block; ***/\*The display property specifies the display behavior (the type of rendering box) of an element. The block variable means it displays an element as a block element (like <p>). It starts on a new line, and takes up the whole width\*/***

transition: 0.1s; ***/\*Specifies how many seconds or milliseconds the transition effect takes to complete. In this case the transition form 0 width to full side bar width\*/***

}

***/\*Positions and designs the menu close button\*/***

.Menu .close { ***/\*selects the close icon within the menu sidebar\*/***

position: absolute; ***/\*The position property specifies the type of positioning method used for an element. The absolute variable means The element is positioned relative to anything it is containted within. In this case the sidebar menu. \*/***

top: 0; ***/\*puts it at the top\*/***

right: 25px; ***/\*puts it that amount from the right\*/***

font-size: 36px;

margin-left: 50px; ***/\*margin creates extra space around an element in this case to the left\*/***

}

.content1 { ***/\*selects the first content section under the image\*/***

text-align: center; /***\*The text-align property specifies the horizontal alignment of text in an element. In this case it is to the center\*/***

position:relative; ***/\*The position property specifies the type of positioning method used for an element. The variable realtive means the element is positioned relative to its normal position, so "left:20px" adds 20 pixels to the element's LEFT position\*/***

}

#title{ ***/\*selects the header\*/***

text-align: center; ***/\*The text-align property specifies the horizontal alignment of text in an element. In this case it is to the center\*/***

font-size: 60px;

color: white;

background-color:#000066;

position:relative; ***/\*The position property specifies the type of positioning method used for an element. The variable realtive means the element is positioned relative to its normal position, so "left:20px" adds 20 pixels to the element's LEFT position\*/***

top:-10%; ***/\* it is moved up relative to where it used to be\*/***

font-family: Arial Black; ***/\*sets the font\*/***

z-index: 1; ***/\*The z-index property specifies the stack order of an element. An element with greater stack order is always in front of an element with a lower stack order.\*/***

}

.back{ ***/\*selects the character of the back button\*/***

z-index: 3; ***/\*The z-index property specifies the stack order of an element. An element with greater stack order is always in front of an element with a lower stack order.\*/***

position: absolute; ***/\*The position property specifies the type of positioning method used for an element. The absolute variable means The element is positioned relative to anything it is containted within. In this case the body. \*/***

top:-5%; ***/\* it is moved up relative to where it used to be\*/***

color: white;

width:50px;***/\*sets the width to 50 pixels\*/***

height:10%; /\****sets the height to 10%\*/***

left:1%; ***/\*it is moved a little form the left\*/***

font-family: Arial; ***/\*sets the font\*/***

}

.backBG{ ***/\*selects the baclground of the back button\*/***

height: 85px; ***/\*it has a height of 85 pixels\*/***

width: 85px; ***/\*it has a width of 85 pixels\*/***

background-color: #4d6e8f;

z-index: 2; ***/\*The z-index property specifies the stack order of an element. An element with greater stack order is always in front of an element with a lower stack order.\*/***

position: absolute; ***/\*The position property specifies the type of positioning method used for an element. The absolute variable means The element is positioned relative to anything it is containted within. In this case the sidebar menu. \*/***

top:-3.4%; ***/\* it is moved up relative to where it used to be\*/***

left:0%; ***/\*it is all the way to the left\*/***

opacity: 0.8;***/\*makes it semi-tranparent\*/***

}

.mySlides {display: none}/***\*hides the slides until it is their turn\*/***

img {vertical-align: middle;} ***/\*puts all the img in the middle from a vertical perspective\*/***

***/\* Slideshow container \*/***

.slideshow-container { ***/\*selector for what holds the presentation\*/***

width: 80%; /***\*sets width to 80%\*/***

position: relative; ***/\*The position property specifies the type of positioning method used for an element. The variable realtive means the element is positioned relative to its normal position, so "left:20px" adds 20 pixels to the element's LEFT position\*/***

right:-12%; ***/\*moves it less the right from where it is automatically positioned\*/***

margin: auto;/***\*centers it\*/***

}

***/\* Next & previous buttons \*/***

.prev, .next { ***/\*selects for the css the next and previous button\*/***

cursor: pointer; ***/\*sets the cursor to the pointer hand while cursor over it\*/***

position: absolute; ***/\*The position property specifies the type of positioning method used for an element. The absolute variable means The element is positioned relative to anything it is containted within. In this case the body. \*/***

top: 50%; ***/\*it ismoved 50% from the top\*/***

width: auto; ***/\*the width isauto done base don what it needs base don font settings\*/***

padding: 16px; ***/\*Padding creates extra space within an element\*/***

margin-top: -22px; ***/\*moves it up from where it usually is\*/***

color: white;

font-weight: bold; ***/\*just like in word it bolds the text\*/***

font-size: 68px;

transition: 0.6s ease; ***/\*sets transition duration and setting\*/***

border-radius: 0 3px 3px 0; ***/\*sets the roundness of the borders on the top, right, bottom, then left\*/***

color:blue;

user-select: none;

}

/***\* Position the "next button" to the right \*/***

.next {

right: 15%; ***/\*moves it 15% from the right\*/***

border-radius: 3px 0 0 3px; ***/\*sets the roundness of the borders on the top, right, bottom, then left\*/***

}

.prev {

left: 15%; ***/moves it 15% from the left\*/***

}

***/\* Fading animation \*/***

.fade {

-webkit-animation-name: fade; ***/\*says what type of animation is happening\*/***

-webkit-animation-duration: 1.5s; ***/\*sets the duration for the animation \*/***

animation-name: fade; ***/\*says what type of animation is happening\*/***

animation-duration: 1.5s; ***/\*sets the duration for the animation \*/***

}

@-webkit-keyframes fade { ***/\*sets the settings for the animation duration\*/***

from {opacity: .4}***/\*it goes from opacity .4\*/***

to {opacity: 1} ***/\*to opacity 1\*/***

}

@keyframes fade { ***/\*sets the settings for the animation duration\*/***

from {opacity: .4} ***/\*it goes from opacity .4\*/***

to {opacity: 1}***/\*to opacity 1\*/***

}

</style>

</head>

<body>

<div id="Menu" class="Menu"> ***<!--sets up the container for the side bar menu-->***

<a href="javascript:void(0)" class="close" onclick="closeMenu()">&times;</a> ***<!-created the close icon that on click actiacted the closeMenu() function. The symbol is created using a char code-->***

<a href="FIRST.html">FIRST</a> ***<!--makes the words take the user to the defind href="" location when clicked-->***

<a href="Team.html">Blue Cheese Robotics</a> ***<!--makes the words take the user to the defind href="" location when clicked-->***

<a href="Compeitions.html">Competitions</a> ***<!--makes the words take the user to the defind href="" location when clicked-->***

<a href="Outreach.html">Our Outreach</a> ***<!--makes the words take the user to the defind href="" location when clicked-->***

</div>

<span style="font-size:40px;cursor:pointer;color:white;font-family: Arial;" onclick="openMenu()">&#9776; Menu</span> ***<!--styles the open menu icon including making the cursor a pointer while above the element, sets the font, and calls the function openMenu() to open the menu when clicked. The wierd charcers infornt of menu makes the three bars symbol.-->***

<script> <!--defines when javascript starts-->

function openMenu() { ***/\*function called when the user clicks on the span with the menu in the corner\*/***

document.getElementById("Menu").style.width = "250px"; ***<!--changes the width to make the side bar slide out assisted with the transition in css-->***

document.getElementById("Menu").style.border = "10px solid #b3b3ff"; ***<!--makes the border around the element come with the sidebar-->***

}

function closeMenu() {

document.getElementById("Menu").style.width = "0"; ***<!--changes the width to make the side bar slide in assisted with the transition in css-->***

document.getElementById("Menu").style.border = "0px"; ***<!--makes it so the border goes away when the side bar goes away-->***

}

</script> ***<!--defines when javascript ends-->***

<div class= "Main\_Content"> ***<!-- container for the actual content of the page-->***

<a href="Team.html"> ***<!--makes everythign inside a link to href=""-->***

<span class="back"style='font-size:85px;'>&#8249;&#8249;</span> ***<!-=creates the back button-->***

<span class="backBG"></span>

</a>

<h1 id="title">Outreach</h1> ***<!--creates the header at the top-->***

<div class="slideshow-container"> ***<!--the container that holds the slideshow-->***

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide1.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide2.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide3.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide4.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide5.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide6.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide7.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide8.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide9.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide10.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide11.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide12.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide13.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide14.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide15.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide16.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide17.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide18.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide19.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide20.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide21.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide22.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide23.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide24.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide25.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide26.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide27.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide28.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide29.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide30.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide31.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide32.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide33.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide34.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide35.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide36.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide37.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide38.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide39.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide40.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide41.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide42.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide43.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide44.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide45.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide46.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide47.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide48.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide49.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide50.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide51.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide52.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide53.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide54.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide55.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide56.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide57.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide58.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide59.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

<div class="mySlides fade"> ***<!--contains the part that will be animated to fade into the next in the presentation-->***

<img src="outreach\_slides/Slide60.png" style="width:70%"> ***<!--inserts and set the width of the indiviual image for the presentation-->***

</div>

</div>

<a class="prev" onclick="plusSlides(-1)">&#10094;</a> ***<!--link that causes the slides to go backwards-->***

<a class="next" onclick="plusSlides(1)">&#10095;</a> ***<!--link that causes the slides to go forwards-->***

</div>

<script>

var slideIndex = 1;***//creates an index to be used in the function that at this moment is one***

showSlides(slideIndex); ***//calls the function putting in the set slideindex of 1***

function plusSlides(n) { ***//when you want to go to the next slide***

showSlides(slideIndex += n);***//this calls the show slides function giving it one number farther to get the slide you want***

}

function currentSlide(n) { ***// when you want to go to the previous slide***

showSlides(slideIndex = n);***//you go to one below this***

}

function showSlides(n) {***//function called in the beggining with n being an inputted*** ***variable uses recursive functions to find where we are supposed to be***

var i; ***//creates a variable i***

var slides = document.getElementsByClassName("mySlides"); ***//creates a variable slides that equal our slides we created in html***

if (n > slides.length) {slideIndex = 1}***// makes it so if the counter n is biiger than the amount of slides it goes back to one***

if (n < 1) {slideIndex = slides.length}***//makes it so if you backwards enough to go past the first you go to the last one***

for (i = 0; i < slides.length; i++) {***//goes through all the slides***

slides[i].style.display = "none"; ***//sets each slide to not display***

}

slides[slideIndex-1].style.display = "block"; ***//sets the slide we want to display***

dots[slideIndex-1].className += " active";***// marks the slide as active, so if we*** ***wanted to do more css in the future we could***

}

</script>

</body>

</html>

## Competitions

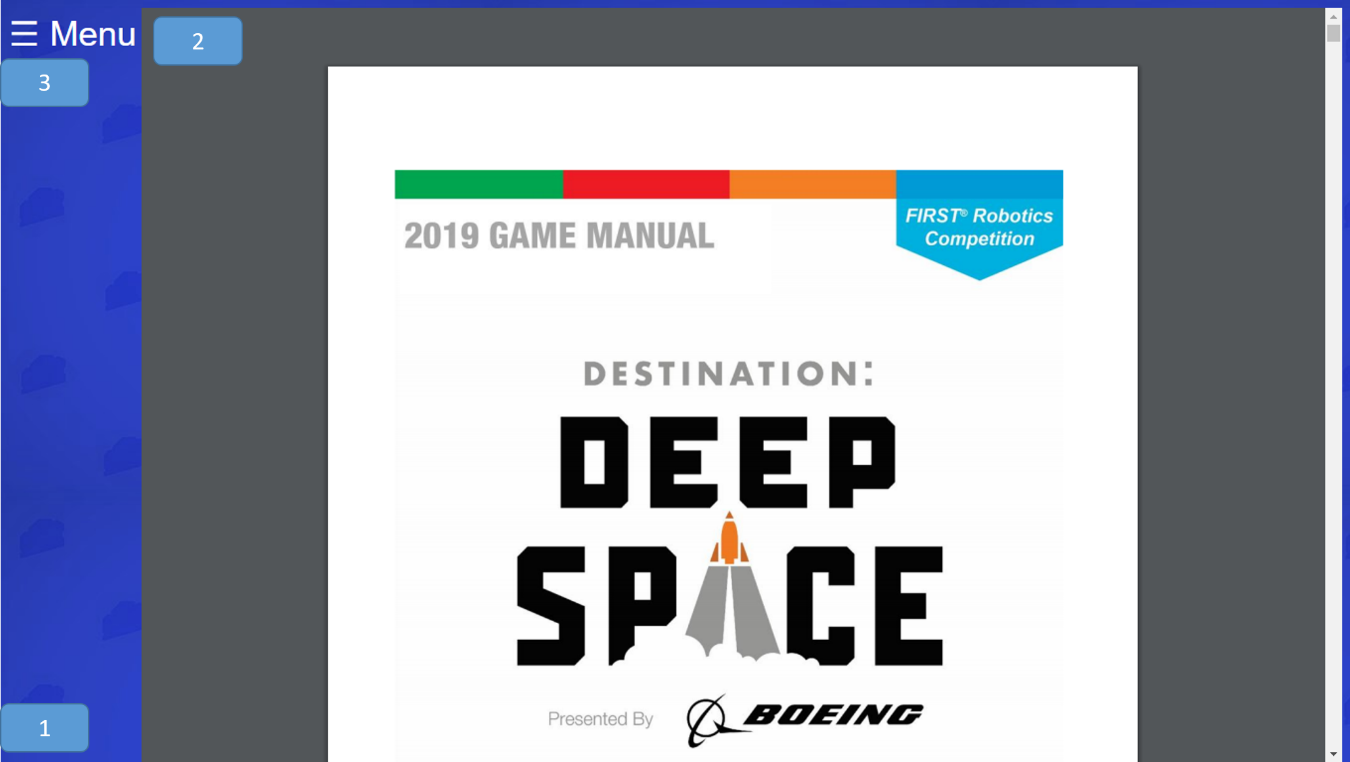
### Overview

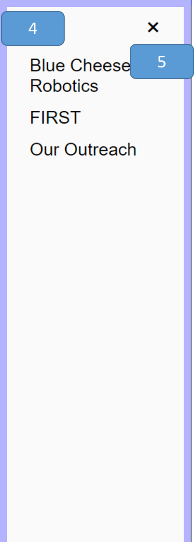
Content: Displays the game manual.

Purpose: In order to provide easy access to the game manual for game participants.

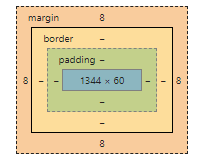
Audience: Student and Mentor users involved deeply involved with the competition.

### Layout

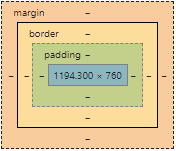




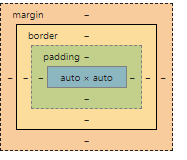
1. body



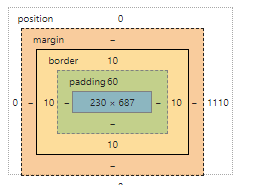
1. object



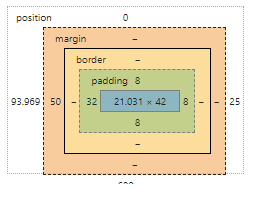
1. Menu



1. Menu



1. close



### Code

<!DOCTYPE html>

<html lang="en"> ***<!--set up that this is an html file and the language is english-->***

<head>

<meta charset="UTF-8"> ***<!--defines the type of charcters being used-->***

<style>

object::-webkit-scrollbar{ ***/\*selects the scroll bar\*/***

width: 50%; ***/\*makes the scroll bar smaller length wise\*/***

}

body { ***/\*effectivly all selector\*/***

background-image: url("bgtile.jpg"); ***/\*sets the background image\*/***

}

***/\*CSS for the side menu bar\*/***

***/\* Set up the side bar menu itself. Javascript will make it so that it will appear by chaning the width from zero \*/***

.Menu { ***/\*selects the overall menu section\*/***

height: 100%; ***/\*sets the height to 100%\*/***

width: 0; ***/\*changed in javascript when you click to open the menu\*/***

position: fixed; ***/\*The position property specifies the type of positioning method used for an element. The fixed variable means that the element is positioned relative to the browser window\*/***

z-index: 4; ***/\*The z-index property specifies the stack order of an element. An element with greater stack order is always in front of an element with a lower stack order.\*/***

top: 0; ***/\*puts it at the top\*/***

left: 0; ***/\*puts it to the left\*/***

background-color: #fafafa;

overflow-x: hidden; ***/\*The overflow-x property specifies whether to clip the content, add a scroll bar, or display overflow content of a block-level element, when it overflows at the left and right edges. The hidden variable makes it so the content is clipped - and no scrolling mechanism is provided\*/***

transition: 0.2s; ***/\*Specifies how many seconds or milliseconds the transition effect takes to complete. In this case the transition form 0 width to full side bar width\*/***

padding-top: 60px; : ***/\*Padding creates extra space within an element on the top\*/***

}

***/\*Design for the words within the side bar menu\*/***

.Menu a { /***\*selects the words within the side bar menu\*/***

padding: 8px 8px 8px 32px; ***/\* sets extra space within the element in the order of top, right, bottom, then left\*/***

text-decoration: none; ***/\*takes away the standard underline in a link\*/***

font-size: 25px;

font-family: Arial;/***\*sets the font\*/***

color: black;

display: block; ***/\*The display property specifies the display behavior (the type of rendering box) of an element. The block variable means it displays an element as a block element (like <p>). It starts on a new line, and takes up the whole width\*/***

transition: 0.1s; ***/\*Specifies how many seconds or milliseconds the transition effect takes to complete. In this case the transition form 0 width to full side bar width\*/***

}

***/\*Positions and designs the menu close button\*/***

.Menu .close { ***/\*selects the close icon within the menu sidebar\*/***

position: absolute; ***/\*The position property specifies the type of positioning method used for an element. The absolute variable means The element is positioned relative to anything it is containted within. In this case the sidebar menu. \*/***

top: 0; ***/\*puts it at the top\*/***

right: 25px; ***/\*puts it that amount from the right\*/***

font-size: 36px;

margin-left: 50px; ***/\*margin creates extra space around an element in this case to the left\*/***

}

</style>

</head>

<body>

<div id="Menu" class="Menu"> ***<!--sets up the container for the side bar menu-->***

<a href="javascript:void(0)" class="close" onclick="closeMenu()">&times;</a> ***<!-created the close icon that on click actiacted the closeMenu() function. The symbol is created using a char code-->***

<a href="Team.html">Blue Cheese Robotics</a***><!--makes the words take the user to the defind href="" location when clicked-->***

<a href="FIRST.html">FIRST</a***><!--makes the words take the user to the defind href="" location when clicked-->***

<a href="Outreach.html">Our Outreach</a***><!--makes the words take the user to the defind href="" location when clicked-->***

</div>

<span style="font-size:35px;cursor:pointer;color:white;font-family: Arial;float:left;" onclick="openMenu()">&#9776; Menu</span> ***<!--styles the open menu icon including making the cursor a pointer while above the element, sets the font, and calls the function openMenu() to open the menu when clicked. The wierd charcers infornt of menu makes the three bars symbol.-->***

<script> <!--defines when javascript starts-->

function openMenu() { ***//function called when the user clicks on the span with the menu in the corner***

***document.getElementById***("Menu").style.width = "250px"; ***//changes the width to make the side bar slide out assisted with the transition in css***

document.getElementById("Menu").style.border = "10px solid #b3b3ff"; /***/makes the border around the element come with the sidebar***

}

function closeMenu() {

document.getElementById("Menu").style.width = "0"; ***<!--changes the width to make the side bar slide in assisted with the transition in css-->***

document.getElementById("Menu").style.border = "0px"; ***<!--makes it so the border goes away when the side bar goes away-->***

}

</script> <!--defines when javascript ends-->

<object data="Competitions/GameManual.pdf" type="application/pdf" height="760px" width="90%" align="right"> ***//creates and object that holds the pdf of***

alt : <a href="Competitions/GameManual.pdf">Click Me</a> ***//makes a link to click on if it is having issues loading***

</object>

</body>

</html>

# Glossary

* <!- - -…- - >
  + Defines a comment
* <a>
  + Defines a hyperlink
* Attribute
  + A value associated with an element, consisting of a name, and an associated (textual) value.
* Body
  + The body is the container for all of a page’s content. Comes after the <head> tag, within the overall <html> tag.
* Border
  + The border shorthand CSS property sets an element's border. It sets the values of border-width, border-style, and border-color.
* <br>
  + Defines a single line break
* <button>
  + Defines a clickable button
* Children
  + An element that is an immediate descendent of another element or nested within another element is called a child. These become useful when using CSS child selectors and psuedo-elements.
* Class
  + HTML elements can have one or more classes, separated by spaces. You can style elements using CSS by selecting them with their classes.
* Comments
  + Information added to the code for future reference. The information may include a brief explanation for what a complex code does. Comments are ignored by the browsers and they are visible in the source code. HTML comments start with <!-- and end with -->.
* Container
  + In HTML, the container is the area enclosed by the beginning and ending tags.
* CSS
  + CSS. Stands for "Cascading Style Sheet." Cascading style sheets are used to format the layout of Web pages. They can be used to define text styles, table sizes, and other aspects of Web pages.
* <data>
  + Links the given content with a machine-readable translation
* <div>
  + Defines a section in a document
* <!-DOCTYPE>
  + Defines the document type
* Element
  + An element in HTML refers to a tag (such as <head>, <body>, and <p>) or element of structure of a document(such as body, title, and paragraph)
* <embed>
  + Defines a container for an external (non-HTML) application
* Font-family
  + The font-family property sets the font of an HTML element’s text.
* <h1> to <h6>
  + Defines HTML headings
* Head
  + Tag that surrounds important content that is invisible to the user, but is important to the browser. Elements within this tag contain metadata about the page and links to stylesheets, scripts, etc.
* Hidden
  + Specifies that an element is not yet, or is no longer, relevant
* HTML
  + HTML stands for Hyper Text Markup Language. It is the language used to create all websites.
* ID
  + An HTML element can have an id attribute to identify it. id elements should always be unique to that single element, and each element should never have more than one id.
* <img>
  + Defines an img
* Inline
  + Elements those that are supported directly by HTML are known as inline. Also, another characteristic of inline element is that their output can be seen or heard without the user taking any additional action (such as clicking, and installing of a plug-in) because the output is directly placed on the webpage. Inline elements include, for instance, animated graphics, graphics, and sound.
* JAVASCRIPT (also see Script)
  + an object-oriented computer programming language commonly used to create interactive effects within web browsers.
* Lang
  + Specifies the language of the element's content
* Margin
  + The margin is the space around the element. The larger the margin, the more space between our element and the elements around it. We can adjust the margin to move our HTML elements closer to or farther from each other. Here, the div with id ‘box’ will get 10px of margin above and below it, and 5px of margin to the left and right.
* Opacity
  + The opacity-level describes the transparency-level, where 1 is not transparent at all, 0.5 is 50% see-through, and 0 is completely transparent.
* <p>
  + Defines a paragraph
* Padding
  + The padding is the spacing between the content and the border (edge of the element.). We can adjust this value with CSS to move the border closer to or farther from the content. Here, the div with id ‘box’ will get 10px of padding all around it.
* Recursive
  + is a function which either calls itself or is in a potential cycle of function calls. As the definition specifies, there are two types of recursive functions. Consider a function which calls itself: we call this type of recursion immediate recursion.
* Property
  + A specific attribute with defined meaning that may be used to describe other resources. A property plus the value of that property for a specific resource is a statement about that resource. A property may define its permitted values as well as the types of resources that may be described with this property.
* Script
  + Defines a client-side script
* Selectors
  + Selectors are used in CSS to select the parts of the HTML that are being styled. You can use several different methods for selecting an element.
* <span>
  + Defines a section in a document
* <style>
  + Defines style information for a document
* SVG
  + Scalable Vector Graphics (SVG) is an XML-based vector image format for two-dimensional graphics with support for interactivity and animation.
* Tag
  + All HTML files live within an over-arching html tag. This is the basic tag that defines an html document.
* <title>
  + Defines a title for the document
* User
  + A user is a person who interacts with a user agent to view, hear, or otherwise use a resource

1. Symbol denotes a folder [↑](#footnote-ref-1)