

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT
THÀNH PHỐ HỒ CHÍ MINH



**MÔN HỌC: TT KIẾN TRÚC VÀ GIAO THỨC IOT
BÁO CÁO CUỐI KỲ**

GVHD: ThS. Trương Quang Phúc

Mã học phần: ITAL328264_23_1_02

Ngày học: Thứ Ba, tiết 1-5

Nhóm sinh viên thực hiện:

Nguyễn Minh Tiên Đạt 21139075

Bùi Kha Lin 21139030

Thành phố Hồ Chí Minh, Tháng 11 năm 2023

MỤC LỤC

MỤC LỤC	1
CHƯƠNG I: TÌM HIỂU VỀ HTML, CSS.....	1
1.1 HTML:	1
1.2 CSS.....	46
II. CHƯƠNG 2: TÌM HIỂU VỀ FIREBASE.....	90
1. Giới thiệu:.....	90
2. Ưu nhược điểm:	90
3. Các bước tạo tài khoản:	91
4. Đọc và giải thích code:	91
CHƯƠNG 3: THIẾT KẾ HỆ THỐNG	96
1. Yêu cầu thiết kế:	96
2. Sơ đồ hệ thống	96
3. Thiết kế phần mềm:	101
4. Kết quả.....	106
Link Web: https://khalin0401.github.io/TT_IoT/	108
Link mô phỏng: https://wokwi.com/projects/382489033222942721	108
Link video Demo: https://youtu.be/kbtIBlmMFtM	108
5. Kết luận:	108
TÀI LIỆU THAM KHẢO	109

CHƯƠNG I: TÌM HIỂU VỀ HTML, CSS

1.1 HTML:

1. Headings	Code	<pre><!DOCTYPE html> <html> <body> <h1>Nối vòng tay lớn</h1> <h2>Rừng núi dang tay nối lại biển xa</h2> <h3>Ta đi vòng tay lớn mãi để nối sơn hà</h3> <h4>Mặt đất bao la anh em ta về</h4> <h5>Gặp nhau mừng như bão cát quay cuồng tròn rộng</h5> <h6>Bàn tay ta nắm nối trọn một vòng Việt Nam</h6> </body> </html></pre>
	Result	<p>Nối vòng tay lớn</p> <p>Rừng núi dang tay nối lại biển xa</p> <p>Ta đi vòng tay lớn mãi để nối sơn hà</p> <p>Mặt đất bao la anh em ta về</p> <p>Gặp nhau mừng như bão cát quay cuồng tròn rộng</p> <p>Bàn tay ta nắm nối trọn một vòng Việt Nam</p>
2. Paragraphs	Code	<pre><!DOCTYPE html> <html> <body> <p>Cờ nối gió đêm vui nói ngày</p> <p>Dòng máu nối con tim đồng loại</p> <p>Dụng tình người trong ngày mới</p> <p> Ghi tùng lum xuống dòng nè xuống quá trời nè mà hiện lên web vẫn đẹp nè </p></pre>

		<pre> <p>Cái này in đậm nè</p> <p><i>Cái này thích nằm nghiên nè</i></p> <P>Cái này <ins>bị gạch chân nè</ins>, còn cái này <sub>nằm ở dưới nè</sub></P> <p>Kiểu mạnh mẽ nè</p> <p><mark>Cái này là hightlight nè</mark></p> <p>Cái này có gạch ngang</p> </body> </html> </pre>
	Result	<p>Cái này in đậm nè</p> <p><i>Cái này thích nằm nghiên nè</i></p> <p>Cái này <u>bị gạch ngang nè</u>, còn cái này nằm ở dưới nè</p> <p>Kiểu mạnh mẽ nè</p> <p>Cái này là</p> <p>Cái này có xóa gì đó</p> <p><u>Inserted</u></p>

5. Quotation	Code	<pre> <!DOCTYPE html> <html> <body> <p>Cái này trích ra từ 1 trang web nè:</p> <blockquote cite="http://www.worldwildlife.org/who/index.html"> Thật đâu dòng với một đồng chữ nè, WWF has worked to help people and nature thrive. As the world's leading conservation organization, WWF works in nearly 100 countries. At every level, we collaborate with people around the world to develop and deliver innovative solutions that protect communities, wildlife, and the places in which they live </blockquote> <p>Câu dẫn này <q>được bỏ trong dấu nháy</q></p> </pre>
--------------	------	---

		<pre><P> Cái này <abbr title="Hiện tên nè">có</abbr> chấm dưới với hiện tên nè</P> </body> </html></pre>
	Result	<p>Cái này trích ra từ 1 trang web nè:</p> <p>Thực đầu dòng với một đồng chữ nè, WWF has worked to help people and nature thrive. As the world's leading conservation organization, WWF works in nearly 100 countries. At every level, we collaborate with people around the world to develop and deliver innovative solutions that protect communities, wildlife, and the places in which they live</p> <p>Câu dẫn này “được bỏ trong đầu nháy”</p> <p>Cái này có chấm dưới với hiện tên nè</p>
6. Colors	Code	<pre><!DOCTYPE html> <html> <body> <h1 style="background-color:rgb(255, 0, 195);">Chữ này có màu nền nè</h1> <p style="background-color:Tomato;">Cái này cũng có màu nền mà khác màu nè </p> <p style="background-color: rgb(134, 123, 10);">Cái này cũng khác màu nữa nè</p> <p style="background-color: rgb(134, 123, 10,0.5);">Cái này giống màu mà lợt hơn 50% nè</p> </p> </body> </html></pre>
	Result	<p>Chữ này có màu nền nè</p> <p>Cái này cũng có màu nền mà khác màu nè</p> <p>Cái này cũng khác màu nữa nè</p> <p>Cái này giống màu mà lợt hơn 50% nè</p>

7. CSS	Code	<pre><!DOCTYPE html> <html> <head> <style> h1 { color: blue; font-family: verdana; font-size: 300%; } p { color: red; font-family: courier; font-size: 160%; border: 3px solid powderblue; } </style> </head> <body> <h1>Cái này dùng thẻ style riêng để định dạng nè</h1> <p>Cái này cũng vậy dùng thẻ riêng để định dạng mà có thêm khung nữa</p> Cái này dùng style bên trong thẻ nè <p style="padding: 30px">Vẫn là nó mà có thêm padding nè - kết hợp thêm style trong thẻ riêng</p> <p style="margin: 100px;">Margin luôn cho thấy sự khác biệt nè</p> <h2>Một tí thông tin thêm về margin, border với padding nè</h2> </body> </html></pre>

		<p>Cái này dùng thẻ style riêng để định dạng nè</p> <p>Cái này cũng vậy dùng thẻ riêng để định dạng mà có thêm khung nữa</p> <p>Cái này dùng style bên trong thẻ nè</p> <p>Vẫn là nó mà có thêm padding nè - kết hợp thêm style trong thẻ riêng</p> <p>Margin luôn cho thấy sự khác biệt nè</p> <p>Một tí thông tin thêm về margin, border với padding nè</p> <p>margin</p> <p>border</p> <p>padding</p> <p>Nội dung</p> <p>codehub.vn</p>
8 . Links	Code	<pre><!DOCTYPE html> <html> <style> a:link { color: green; background-color: transparent; text-decoration: none; } a:visited { color: pink; background-color: transparent; text-decoration: none; } a:hover { color: red; background-color: transparent; text-decoration: underline; } a:active { color: yellow; background-color: transparent; text-decoration: underline; } </style> <body> <h2> Thẻ a và lệnh href dùng để chèn link vào nè</h2></pre>

```
<a href="https://www.w3schools.com/" target="_blank" id="C1">Tên  
linkkk, có thêm target="_blank" thì nó sẽ mở ở tab mới nè</a>  
  
<h3>Cũng hiện link mà có thêm title</h3>  
<p>Nó sẽ hiện vài thông tin nhỏ khi chuyên chuột vào nè</p>  
<a href="https://www.w3schools.com/html/" title="Hiện lên nè ảo  
chưa">Visit our HTML Tutorial</a>  
  
<h3>Cũng hiện link mà dùng bằng hình ảnh nè</h3>  
<a href="https://www.w3schools.com/html/">  
      
    </a>  
<h3>Cũng là link mà nó có đổi màu nè</h3>  
<a href="https://www.w3schools.com/">Link nè</a>  
<p>Xuống vài dòng</p>  
<p><a href="#C1">nhảy về đầu trang</a></p>  
</body>  
</html>
```

		<p>Thẻ a và lệnh href dùng để chèn link vào nè</p> <p>Tên linkkk, có thêm target="_blank" thì nó sẽ mở ở tab mới nè</p> <p>Cũng hiện link mà có thêm title</p> <p>Nó sẽ hiện vài thông tin nho nhỏ khi chuột vào nè</p> <p>Visit our HTML Tutorial</p> <p>Cũng hiện link mà dùng bằng hình ảnh nè</p>  <p>Cũng là link mà nó có đổi màu nè</p> <p>Link nè</p> <p>Xuống vài dòng</p> <p>nhảy về đầu trang</p>
9. Image	Code	<pre><!DOCTYPE html> <html> <style></pre>

```



</style>



<body>



<h2>Chèn ảnh bình thường nè dùng src đặt tên link, alt để đặt tên, alt không bị ng dùng nhìn thấy</h2>



<h2>Khi không có đặt kích thước cụ thể cho từng ảnh thì nó sẽ lấy đặc trưng chung của img ở trên là 100% width ảnh hơi to xíu ^^</h2>







<p>Nếu mà không tìm thấy dc hình ảnh thì nó sẽ hiện alt</p>







<p>Kết hợp với điều khiển size nè</p>







<p style="text-align: right;"> Lấy img từ 1 sever nè và định dạng sang  

phải</p>







<p>Cái này là ảnh gif</p>







<p>Chèn link vào ảnh</p>



<a href="https://online.hcmute.edu.vn/">  

</a>



<p>



Chèn ảnh chung với chữ nè







</p>


```

		<pre></body> </html></pre>
	<p>Result</p>	<p>Chèn ảnh bình thường nè dùng src đặt tên link, alt để đặt tên, alt không bị ng dùng nhìn thấy</p> <p>Khi không có đặt kích thước cụ thể cho từng ảnh thì nó sẽ lấy đặc trưng chung của img ở trên là 100% width</p>  <p>Nếu mà không tin thấy dc hình ảnh thi nó sẽ hiện alt</p> <p>Không ra được ảnh nè do src sai</p> <p>Kết hợp với điều khiển size nè</p>  <p>Cái này là ảnh gif</p> <p></p> <p>Chèn link vào ảnh</p> <p></p> <p> Chèn ảnh chung với chữ nè</p> <p>Lấy img từ 1 sever nè</p> 
10.1 Table Borders	Code	<pre><html> <style> table, th, td { border:1px solid black; } </style> <body></pre>

<h2>1. Sử dụng "tr" cho mỗi hàng của bảng nè, "td" cho mỗi ô của hàng</h2>

```
<table style="width:100%">
<tr>
  <td>Ô1</td>
  <td>Ô2 nè</td>
  <td>Ô3 nhoa</td>
</tr>
</table>
```

<h2>2. TH đại diện cho header nè, tức là in đậm và to hơn đó</h2>

```
<table style="width:100%">
  <caption>Có thêm caption nè, kiểu tiêu đề á</caption>
<tr>
  <th>Header nè</th>
  <th>Thấy to hơn không</th>
  <th>in đậm nữa</th>
</tr>
<tr>
  <td>còn này là nội dung</td>
  <td>blabla</td>
  <td>blabla nè</td>
</tr>
<tr>
  <td>lại là nội dung</td>
  <td>nội dung</td>
  <td>nội dung cuối</td>
</tr>
</table>

</body>
</html>
```

1. Sử dụng "tr" cho mỗi hàng của bảng nè, "td" cho mỗi ô của hàng

Ô1	Ô2 nè	Ô3 nhoa
----	-------	---------

2. TH đại diện cho header nè, tức là in đậm và to hơn đó

Có thêm caption nè, kiểu tiêu đề á

Header nè	Thấy to hơn không	in đậm nữa
còn này là nội dung	blabla	blabla nè
lại là nội dung	nội dung	nội dung cuối

Result

Code

```
<!DOCTYPE html>
```

10.2 HTML Table Sizes

```
<html>
<style>
table, th, td {
    border:1px solid black;
    border-collapse: collapse;
}
</style>
<body>

<h2>1. định dạng độ rộng của cột bằng width</h2>

<table style="width:100%">
    <tr>
        <th style="width:100%">Cái này để 100% chiều ngang nè, nhưng nó
vẫn sẽ chừa chỗ cho các cột còn lại nếu có</th>
        <th>Lastname</th>
        <th>Age</th>
    </tr>
    <tr>
        <td>Jill</td>
        <td>Smith</td>
        <td>50</td>
    </tr>
</table>

<table style="width:100%">
    <tr>
        <th style="width:70%">Cái này để 70% chiều ngang nè</th>
        <th>Lastname</th>
        <th>Age</th>
    </tr>
    <tr>
        <td>Jill</td>
        <td>Smith</td>
        <td>50</td>
    </tr>
</table>

<h2>2. Phần này là set height của hàng nè </h2>

<table style="width:100%">
    <tr>
        <th>Firstname</th>
        <th>Lastname</th>
        <th>Age</th>
```

		<pre> </tr> <tr style="height:200px"> <td>Cái này là set 200px</td> <td>Smith</td> <td>50</td> </tr> <tr style="height:400px"> <td>Cái này set h&gt;n 400px luôn nha</td> <td>Smith</td> <td>50</td> </tr> </body> </html> </pre>																									
	Result	<p>1. Định dạng độ rộng của cột bằng width</p> <table border="1"> <thead> <tr> <th colspan="2">Cái này để 100% chiều ngang nè, nhưng nó vẫn sẽ chừa chỗ cho các cột còn lại nếu có</th> <th>Lastname</th> <th>Age</th> </tr> </thead> <tbody> <tr> <td>Jill</td> <td></td> <td>Smith</td> <td>50</td> </tr> <tr> <th colspan="2">Cái này để 70% chiều ngang nè</th> <th>Lastname</th> <th>Age</th> </tr> <tr> <td>Jill</td> <td></td> <td>Smith</td> <td>50</td> </tr> </tbody> </table> <p>2. Phần này là set height của hàng nè</p> <table border="1"> <thead> <tr> <th>Firstname</th> <th>Lastname</th> <th>Age</th> </tr> </thead> <tbody> <tr> <td>Cái này là set 200px</td> <td>Smith</td> <td>50</td> </tr> <tr> <td>Cái này set h&gt;n 400px luôn nha</td> <td>Smith</td> <td>50</td> </tr> </tbody> </table>	Cái này để 100% chiều ngang nè, nhưng nó vẫn sẽ chừa chỗ cho các cột còn lại nếu có		Lastname	Age	Jill		Smith	50	Cái này để 70% chiều ngang nè		Lastname	Age	Jill		Smith	50	Firstname	Lastname	Age	Cái này là set 200px	Smith	50	Cái này set h>n 400px luôn nha	Smith	50
Cái này để 100% chiều ngang nè, nhưng nó vẫn sẽ chừa chỗ cho các cột còn lại nếu có		Lastname	Age																								
Jill		Smith	50																								
Cái này để 70% chiều ngang nè		Lastname	Age																								
Jill		Smith	50																								
Firstname	Lastname	Age																									
Cái này là set 200px	Smith	50																									
Cái này set h>n 400px luôn nha	Smith	50																									
10.3 Table Headers	Code	<pre> <!DOCTYPE html> <html> <head> <style> table, th, td { </pre>																									

```

border: 1px solid black;
border-collapse: collapse;
}
</style>
</head>
<body>

<p>Sử dụng th để đánh dấu header</p>

<table style="width:100%">
<tr>
<th>To hơn</th>
<th>dậm hơn</th>
<th>Nỗi bậc hơn</th>
</tr>
<tr>
<td>Jill</td>
<td>Smith</td>
<td>50</td>
</tr>
</table>

<h2>Cũng là sử dụng th mà theo một cách khác nè</h2>
<table style="width:100%">
<tr>
<th>Nó còn được tình bày theo kiểu này nè</th>
<td>Jill</td>
<td>Eve</td>
</tr>
<tr>
<th>Vẫn sẽ nỗi bậc hơn</th>
<td>Smith</td>
<td>Jackson</td>
</tr>
</table>
<h2>Còn cái này dùng text-align để định dạng cho chữ nằm trong ô
nè</h2>
<table style="width:100%">
<tr style="text-align: center;">
<th>Chữ nằm ở giữa nè</th>
<th>Cái này cũng dc canh giữa</th>
<th>này nữa</th>
</tr>
<tr>
<td>Jill</td>

```

```

<td>Smith</td>
<td>50</td>
</tr>
</table>

<p>3. Cái này sử dụng colspan để gộp 2 ô thành 1 nè</p>

<table style="width:100%">
<tr>
<th colspan="2" style="background-color: aqua;">Ô được gộp</th>
<th>Age</th>
</tr>
<tr>
<td>Jill</td>
<td>Smith</td>
<td>50</td>
</tr>
<tr>
<td>Eve</td>
<td>Jackson</td>
<td>94</td>
</tr>
</table>
</body>
</html>

```

Sử dụng th để đánh dấu header

To hơn	dậm hơn	Nội bặc hơn
Jill	Smith	50

Cũng là sử dụng th mà theo một cách khác nè

Nó còn được tính bày theo kiểu này nè	Jill	Eve
Vẫn sẽ nội bặc hơn	Smith	Jackson

Result

Còn cái này dùng text-align để định dạng cho chữ nằm trong ô nè

Chữ nằm ở giữa nè	Cái này cũng dc canh giữa	này nưa
Jill	Smith	50

3. Cái này sử dụng colspan để gộp 2 ô thành 1 nè

Ô được gộp	Age
Jill	50
Eve	94

10.4 Padding

Code

```

<!DOCTYPE html>
<html>
<head>
<style>
table, th, td {
  border: 1px solid black;
  border-collapse: collapse;
}

```

```

th, td {
    padding-top: 10px;
    padding-bottom: 20px;
    padding-left: 30px;
    padding-right: 40px;
}
</style>
</head>
<body>

<p>Sử dụng padding để thay đổi kích thước của padding nè</p>

<table style="width:100%">
<tr>
    <th>Firstname</th>
    <th>Lastname</th>
    <th>Age</th>
</tr>
<tr>
    <td>Jill</td>
    <td>Smith</td>
    <td>50</td>
</tr>
<tr>
    <td>Eve</td>
    <td>Jackson</td>
    <td>94</td>
</tr>
</table>

<p><strong>Tip:</strong> Try to change the padding to 5px.</p>

</body>
</html>

```

Sử dụng padding để thay đổi kích thước của padding nè

Firstname	Lastname	Age
Jill	Smith	50
Eve	Jackson	94

Tip: Try to change the padding to 5px.

11. Lists

Code

```
<!DOCTYPE html>
<html>
```

		<pre> <body> <h2>Một số cách biểu diễn list trong html</h2> Sử dụng ul để tạo ra chấm tròn màu đen li để biểu diễn các nội dung trong list Còn ol để thể hiện bằng dấu gạch đầu dòng Vẫn sử dụng li <p style="color: brown;">Còn có thể sử dụng dl để định nghĩa 1 cái list</p> <dl> <dt>Dt để tạo một mục lớn</dt> <dd>- dd để tạo một mục nhỏ hơn</dd> <dt>Milk</dt> <dd>- white cold drink</dd> </dl> </body> </html> </pre>
Result		<p>Một số cách biểu diễn list trong html</p> <ul style="list-style-type: none"> Sử dụng ul để tạo ra chấm tròn màu đen li để biểu diễn các nội dung trong list <ol style="list-style-type: none"> Còn ol để thể hiện bằng dấu gạch đầu dòng Vẫn sử dụng li <p>Còn có thể sử dụng dl để định nghĩa 1 cái list</p> <p>dt để tạo một mục lớn</p> <ul style="list-style-type: none"> - dd để tạo một mục nhỏ hơn <p>Milk</p> <ul style="list-style-type: none"> - white cold drink
12. Block & Inline	Code	<pre> <!DOCTYPE html> <html> <body> </pre>

```

<h2>Thay đổi kiểu dấu đầu dòng trong định dạng ul bằng list-style-type
</h2>

<ul>
  <li>Này thì nó bình thường</li>
  <li>Tròn</li>
  <li>In đậm</li>
</ul>

<ul style="list-style-type:disc;">
  <li>Cái này được định dạng thêm dạng disc</li>
  <li>Cũng là tròn</li>
  <li>đậm</li>
</ul>

<ul style="list-style-type:circle;">
  <li>Cái này được định dạng thêm dạng circle</li>
  <li>tròn</li>
  <li>rỗng</li>
</ul>

<ul style="list-style-type:square;">
  <li>Cái này được định dạng thêm dạng square</li>
  <li>vuông</li>
  <li>đem</li>
</ul>

<ul style="list-style-type:none;">
  <li>Cái này được định dạng thêm dạng none</li>
  <li>không có ký hiệu đầu dòng nè</li>
</ul>

<p>Sau đây là kiểu định dạng theo thứ tự của list nè</p>
<ol type="i">
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ol>
<ol type="I">
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ol>

<ol type="A">

```

		<pre> Coffee Tea Milk <ol type="1"> Coffee Tea Milk </body> </html> </pre>
	Result	<p>Thay đổi kiểu dấu đầu dòng trong định dạng ul bằng list-style-type</p> <ul style="list-style-type: none"> • Nay thi nó bình thường • Tròn • In đậm <ul style="list-style-type: none"> Cái này được định dạng thêm dạng disc Cũng là tròn đậm <ul style="list-style-type: none"> Cái này được định dạng thêm dạng circle tròn rỗng <ul style="list-style-type: none"> Cái này được định dạng thêm dạng square vuông đem <p>Cái này được định dạng thêm dạng none không có ký hiệu đầu dòng nè</p> <p>Sau đây là kiểu định dạng theo thứ tự của list nè</p> <ul style="list-style-type: none"> i. Coffee ii. Tea iii. Milk <ul style="list-style-type: none"> I. Coffee II. Tea III. Milk <ul style="list-style-type: none"> A. Coffee B. Tea C. Milk <ul style="list-style-type: none"> 1. Coffee 2. Tea 3. Milk
13. Iframes	Code	<pre> <!DOCTYPE html> <html> <body> <h2>Chèn iframe vào nè, sử dụng thẻ iframe và sử dụng đường link nha</h2> <iframe src="https://puppyhouse.vn/wp-content/uploads/2019/11/kha-nang-cua-meo-5-900x598.jpg" name="iframe_a" height="300px" width="100%" title="Iframe Example"></iframe> </pre>

		<pre><p>Bấm vào đây sẽ thay đổi if vì nó được liên kết bằng name</p> </body> </html></pre>
	Result	<p>Chèn iframe vào nè, sử dụng thẻ iframe và sử dụng đường link nha</p>  <p><u>Bấm vào đây sẽ thay đổi if vì nó được liên kết bằng name</u></p>
14. JavaScript	Code	<pre><!DOCTYPE html> <html> <body> <h1>1. Ví dụ số 1 cho việc sử dụng JavaScript</h1> <p>JavaScript có thể thay đổi nội dung của phần tử trong HTML</p> <button type="button1" onclick="myFunction1()">Sử dụng thẻ button để tạo 1 nút nhấn và có kèm thêm thuộc tính onclick kích hoạt hàm myFunction</button> <p id="demo1">Cái này là nội dung lúc đầu có id là demo</p> <script> function myFunction1() { document.getElementById("demo1").innerHTML = "Đây là nội dung sau khi được thay đổi, sử dụng hàm .innerHTML để thay đổi"; } </script> <h1>2. JavaScrip còn có thể thay đổi định dạng</h1> <p id="demo2">Đây là dòng sẽ được định dạng lại, nhớ thêm id cho thẻ</p> <script></pre>

```

function myFunction() {
    document.getElementById("demo2").style.fontSize = "25px";
    document.getElementById("demo2").style.color = "red";
    document.getElementById("demo2").style.backgroundColor = "yellow";
}
</script>

<button type="button2" onclick="myFunction()">Bấm vào đây để kích
hoạt hàm thay đổi 3 định dạng: fontSize, color,
backgroundColor</button>

<h1>3. Nó còn có thể thay đổi hình ảnh </h1>

<script>

function light(sw) {
    var pic;
    if (sw == 0) {
        pic = "https://png.pngtree.com/png-vector/20190903/ourlarge/pngtree-
light-bulb-icon-vector-light-bulb-ideas-symbol-illustration-png-
image_1722195.jpg"
    } else {
        pic = "https://png.pngtree.com/png-vector/20190815/ourlarge/pngtree-
light-bulb-icon-vector--light-bulb-ideas-symbol-illustration-png-
image_1692654.jpg"
    }
    document.getElementById('myImage').src = pic;
}
</script>



<p>
<button type="button" onclick="light(1)">Light On</button>
<button type="button" onclick="light(0)">Light Off</button>
</p>

</body>
</html>

```

Result

Đây là trước khi bấm vào button

1. Ví dụ số 1 cho việc sử dụng JavaScript

JavaScript có thể thay đổi nội dung của phần tử trong HTML

Sử dụng thẻ button để tạo 1 nút nhấn và có kèm thêm thuộc tính onclick kích hoạt hàm myFunction

Cái này là nội dung lúc đầu có id là demo

2. JavaScript còn có thể thay đổi định dạng

Đây là dòng sẽ được định dạng lại, nhớ thêm id cho thẻ

Bấm vào đây để kích hoạt hàm thay đổi 3 định dạng: fontSize, color, backgroundColor

3. Nó còn có thể thay đổi hình ảnh



Light On Light Off

Đây là sau khi bấm vào button

1. Ví dụ số 1 cho việc sử dụng JavaScript

JavaScript có thể thay đổi nội dung của phần tử trong HTML

Sử dụng thẻ button để tạo 1 nút nhấn và có kèm thêm thuộc tính onclick kích hoạt hàm myFunction

Đây là nội dung sau khi được thay đổi, sử dụng hàm .innerHTML để thay đổi

2. JavaScript còn có thể thay đổi định dạng

Đây là dòng sẽ được định dạng lại, nhớ thêm id cho thẻ

Bấm vào đây để kích hoạt hàm thay đổi 3 định dạng: fontSize, color, backgroundColor

3. Nó còn có thể thay đổi hình ảnh

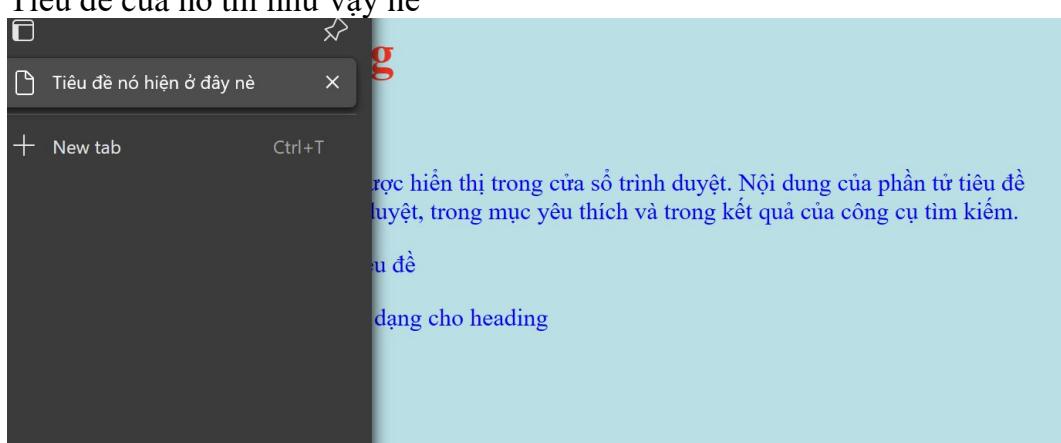


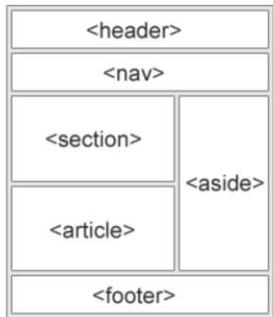
Light On Light Off

15. Head

Code

```
<!DOCTYPE html>
<html>
<head>
```

	<pre> <base href="https://www.w3schools.com/" target="_blank"> </head> <body> <h1>The base element</h1> <p> - Notice that we have only specified a relative address for the image. Since we have specified a base URL in the head section, the browser will look for the image at "https://www.w3schools.com/images/stickman.gif".</p> <p>HTML base tag - Notice that the link opens in a new window, even if it has no target="_blank" attribute. This is because the target attribute of the base element is set to "_blank".</p> </body> </html> </pre>
Result	<p>Đây là heading</p> <p>Đây là paragraph</p> <p>Nội dung của phần tử body được hiển thị trong cửa sổ trình duyệt. Nội dung của phần tử tiêu đề được hiển thị trong tab trình duyệt, trong mục yêu thích và trong kết quả của công cụ tìm kiếm.</p> <p>Sử dụng thẻ title để cài đặt tiêu đề</p> <p>Có thể sử dụng thẻ style định dạng cho heading</p> <p>Tiêu đề của nó thì như vậy nè</p> 

16. Layout	Code	<p>Một ít thông tin thêm cho layout:</p> <h2>Các yếu tố bố cục HTML</h2> <p>HTML có một số yếu tố ngữ nghĩa xác định các phần khác nhau của một trang web:</p>  <ul style="list-style-type: none"> • <header> - Xác định tiêu đề cho một tài liệu hoặc một phần • <nav> - Xác định một tập hợp các liên kết điều hướng • <section> - Xác định một phần trong tài liệu • <article> - Định nghĩa độc lập, khép kín nội dung • <aside> - Xác định nội dung ngoài nội dung (như thanh bên) • <footer> - Xác định chân trang cho một tài liệu hoặc một phần • <details> - Xác định chi tiết bổ sung mà người dùng có thể mở và đóng theo yêu cầu • <summary> - Định nghĩa một tiêu đề cho phần tử <details> <p>Bạn có thể đọc thêm về các yếu tố ngữ nghĩa trong chương HTML Semantics của chúng tôi.</p> <pre data-bbox="458 677 1486 1936">Code: <!DOCTYPE html> <html lang="en"> <head> <title>CSS Template</title> <meta charset="utf-8"> <meta name="viewport" content="width=device-width, initial-scale=1"> <style> * { box-sizing: border-box; } body { font-family: Arial, Helvetica, sans-serif; } /* Định dạng cho header */ header { background-color: #666; padding: 30px; text-align: center; font-size: 35px; color: white; } /* Create two columns/boxes that floats next to each other */ nav { float: left; width: 30%; height: 300px; /* only for demonstration, should be removed */ background: #ccc; padding: 20px; }</pre>
------------	------	---

```

/* Style the list inside the menu */
nav ul {
    list-style-type: none;
    padding: 0;
}

article {
    float: left;
    padding: 20px;
    width: 70%;
    background-color: #f1f1f1;
    height: 300px; /* only for demonstration, should be removed */
}

/* Clear floats after the columns */
section::after {
    content: "";
    display: table;
    clear: both;
}

/* Style the footer */
footer {
    background-color: #777;
    padding: 10px;
    text-align: center;
    color: white;
}

/* Responsive layout - makes the two columns/boxes stack on top of each
   other instead of next to each other, on small screens */
@media (max-width: 600px) {
    nav, article {
        width: 100%;
        height: auto;
    }
}

</style>
</head>
<body>

<h2>Phần này là header</h2>
<p>Trong ví dụ này, chúng tôi đã tạo tiêu đề, hai cột/hộp và chân trang.
Trên màn hình nhỏ hơn, các cột sẽ xếp chồng lên nhau</p>

```

		<pre> <header> <h2>Phần header</h2> </header> <section> <nav> Phía dưới là nav Paris Tokyo </nav> <article> <h1>Cột thứ 2</h1> </article> <footer> <p>Phần cuối nè</p> </footer> </body> </html> </pre>
Result		<p>Phần này là header</p> <p>Trong ví dụ này, chúng tôi đã tạo tiêu đề, hai cột/hộp và chân trang. Trên màn hình nhỏ hơn, các cột sẽ xếp chồng lên nhau</p>
17. Symbols	Code	<!DOCTYPE html> <html>

		<pre> <body> <h1> Nhiều ký hiệu toán học, kỹ thuật và tiền tệ không có trên bàn phím thông thường. Để thêm các ký hiệu như vậy vào trang HTML, bạn có thể sử dụng Symbols có sẵn của HTML. </h1> <p>Gọi bằng tên tiếng anh &euro;</p> <p>Gọi bằng số ký tự thập pahan &#8364;</p> <p>Gọi bằng mã hex &#x20AC;</p> </body> </html> </pre>
	Result	<p>Nhiều ký hiệu toán học, kỹ thuật và tiền tệ không có trên bàn phím thông thường.</p> <p>Để thêm các ký hiệu như vậy vào trang HTML, bạn có thể sử dụng tên thực thẻ hoặc số thực thẻ (tham chiếu thập phân hoặc thập lục phân) cho ký hiệu.</p> <p>Gọi bằng tên tiếng anh €</p> <p>Gọi bằng số ký tự thập pahan €</p> <p>Gọi bằng mã hex €</p>
18. Emojis	Code	<pre> <!DOCTYPE html> <html> <head> <meta charset="UTF-8"> </head> <body> <h1>Emojis cũng được HTML cung cấp sẵn chúng ta có thể gọi nó bằng cách sử dụng &# + số hiệu của Emojis muốn gọi</h1> <p>Tôi sẽ gọi A B C bằng emojis</p> <p>I will display &#65; &#66; &#67;</p> <p>Một ký tự hình ảnh</p> <p>&#128512;</p> <p style="font-size:48px"> Kết hợp thêm style &#128512; &#128516; &#128525; &#128151; </p> </pre>

		</body> </html>
	Result	<p>Emojis cũng được HTML cung cấp sẵn chúng ta có thể gọi nó bằng cách sử dụng &# + số hiệu của Emojis muốn gọi</p> <p>Tôi sẽ gọi A B C bằng emojis</p> <p>I will display A B C</p> <p>Một ký tự hình ảnh</p> <p>😊</p> <p>Kết hợp thêm style</p>  
19. URL Encode	Result	<p>Bộ định vị tài nguyên thống nhất HTML</p> <p>◀ Trước</p> <p>Kế tiếp ▶</p> <p>URL là một từ khác cho một địa chỉ web.</p> <p>URL có thể bao gồm các từ (ví dụ: w3schools.com) hoặc địa chỉ Giao thức Internet (IP) (ví dụ: 192.68.20.50).</p> <p>Hầu hết mọi người nhập tên khi lướt web, bởi vì tên dễ nhớ hơn số.</p> <p>URL - Bộ định vị tài nguyên thống nhất</p> <p>Trình duyệt web yêu cầu các trang từ máy chủ web bằng cách sử dụng URL.</p> <p>Bộ định vị tài nguyên thống nhất (URL) được sử dụng để giải quyết tài liệu (hoặc dữ liệu khác) trên web.</p> <p>Một địa chỉ web như https://www.w3schools.com/html/default.asp tuân theo các quy tắc cú pháp sau:</p> <pre>scheme://prefix.domain:port/path/filename</pre> <p>Lời giải thích:</p> <ul style="list-style-type: none"> • Lực đồ - xác định loại dịch vụ Internet (phổ biến nhất là http hoặc https) • Tiền tố - xác định tiền tố miền (mặc định cho HTTP là www) • domain - định nghĩa tên miền Internet (như w3schools.com) • cổng - xác định số cổng tại máy chủ (mặc định cho http là 80) • path - xác định đường dẫn tại máy chủ (Nếu bỏ qua: thư mục gốc của trang web) • Tên tệp - xác định tên của tài liệu hoặc tài nguyên
20. Forms	Code	<!DOCTYPE html>

```

<html>
<body>

<h2>1. HTML Forms sử dụng để thu thập thông tin</h2>

<form>
    <p>Để tạo được 1 ô nhập dữ liệu cần có 1 nhãn label (có mã for bắng tên id của input) và 1 nhãn input </p>
    <label for="fname">First name:</label><br>
    <input type="text" id="fname" name="fname" value="Nhập ở đây nè"><br>
    <label for="lname">Last name:</label><br>
    <input type="text" id="lname" name="lname" value="Nhập ở đây nữa"><br><br>
</form>

<h2>2. Ví dụ cho dữ liệu ô dạngadio</h2>
<form>
    <input type="radio" id="html" name="fav_language" value="HTML">
    <label for="html">check vào đây được nè</label><br>
    <input type="radio" id="css" name="fav_language" value="CSS">
    <label for="css">đây nữa</label><br>
    <input type="radio" id="test" name="fav_language" value="CSS">
    <label for="css">nhung dạng radio thì chỉ check vào được 1 ô thôi nha</label><br>
</form>

<h2>3. Ví dụ cho dữ liệu ô dạngcheckbox - có thể chọn nhiều tick</h2>
<form>
    <input type="checkbox" id="vehicle1" name="vehicle1" value="Bike">
    <label for="vehicle1">có thể check vào đây nha</label><br>
    <input type="checkbox" id="vehicle2" name="vehicle2" value="Car">
    <label for="vehicle2">trong lúc đó có thể chọn cả 2 luôn</label><br>
    <input type="checkbox" id="vehicle3" name="vehicle3" value="Boat">
    <label for="vehicle3">tham thêm tí thì chọn cả 3 luôn</label><br><br>
</form>

<h2>4. Định dạng submit sẽ thực hiện action đã được khai báo</h2>

<form action="/luu.php">
    <label for="fname">First name:</label><br>
    <input type="text" id="fname" name="fname" value="John"><br>
    <label for="lname">Last name:</label><br>
    <input type="text" id="lname" name="lname" value="Doe"><br><br>

```

```
<input type="submit" value="Bấm vào đây để thực hiện hành động">
```

```
</form>
```

<h2>5. Chọn sử dụng dữ liệu có sẵn - sử dụng for và select có id giống với for</h2>

```
<form >
```

```
  <label for="cars">Chọn 1 nè:</label>
```

```
  <select id="cars" name="cars">
```

```
    <option value="volvo">Helu</option>
```

```
    <option value="saab">Húke</option>
```

```
    <option value="fiat">Hélu</option>
```

```
    <option value="audi" selected>Luhe</option> <!-- Thêm selected để chọn trước-->
```

```
  </select>
```

```
</form>
```

<h2>6. Cài thêm thuộc tính multiple để có thể chọn nhiều giá trị</h2>

```
<form action="/action_page.php">
```

```
  <label for="cars">Chọn được nhiều nè:</label>
```

```
  <select id="cars" name="cars" multiple>
```

```
    <option value="volvo">Volvo</option>
```

```
    <option value="saab">Saab</option>
```

```
    <option value="fiat">Fiat</option>
```

```
    <option value="audi">Audi</option>
```

```
  </select>
```

```
</form>
```

<h2>7. Dùng textarea để nhập được văn bản nhiều dòng</h2>

```
<form>
```

```
  <textarea name="message" rows="10" cols="30">Nhập nhiều dòng vào đây được nè.</textarea>
```

```
</form>
```

<h2>8. Gom các phần tử quan trọng lại</h2>

<p>Phân tử fieldset được sử dụng để nhóm dữ liệu liên quan trong một biểu mẫu. Phân tử legend xác định chú thích cho phân tử fieldset.</p>

```
<form >
```

```
  <fieldset>
```

```
    <legend>Tiêu đề chính sử dụng nhãn legend nè:</legend>
```

```

<label for="fname">First name:</label><br>
<input type="text" id="fname" name="fname" value="John"><br>
<label for="lname">Last name:</label><br>
<input type="text" id="lname" name="lname" value="Doe"><br><br>
</fieldset>
</form>

<h2>9. Sử dụng nhãn datalist để hiện list</h2>

<form>
<input list="browsers" name="browser">
<datalist id="browsers">
<option value="Internet Explorer">
<option value="Firefox">
<option value="Chrome">
<option value="Opera">
<option value="Safari">
</datalist>
</form>

<h2>10. Sử dụng input với 2 dạng range và number </h2>

<form oninput="x.value=parseInt(a.value)+parseInt(b.value)">
0
<input type="range" id="a" name="a" value="50">
100 +
<input type="number" id="b" name="b" value="50">
=
<output name="x" for="a b"></output>
<br><br>
<input type="submit">
</form>

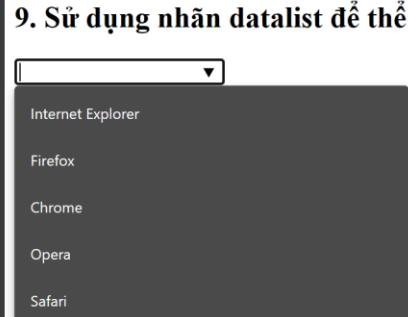
<h1>11. tải file lên</h1>

<form action="/action_page.php">
<label for="myfile">Nó sẽ hiện lên 1 hộp thoại để mở file</label>
<input type="file" id="myfile" name="myfile"><br><br>
<input type="submit" value="Submit">
</form>

<h2>12. Một số thuộc tính thêm của Form</h2>
<p>- Thuộc tính autocomplete sẽ cho khả năng trình duyệt tự dự đoán
diễn dựa trên các thông tin trước của người dùng</p>

```

	<p><input type="text"> - Thuộc tính readonly không cho người dùng khả năng chỉnh sửa nhưng có thể tích vào để copy</p></p> <p><input type="disabled">- Thuộc tính disabled đầu vào chỉ định rằng trường đầu vào sẽ bị vô hiệu hóa. Trường nhập bị tắt sẽ không sử dụng được và không thể nhấp vào được. Giá trị của trường nhập bị tắt sẽ không được gửi khi gửi biểu mẫu!</p></p> <p><input type="multiple">- Thuộc tính multiple chỉ định rằng người dùng được phép nhập nhiều giá trị vào trường đầu vào</p></p> <p><input type="url">- Thuộc tính formaction chỉ định URL của tệp sẽ xử lý đầu vào khi biểu mẫu được gửi</p></p> <p></body></p> <p></html></p>
Result	<p>1. HTML Forms sử dụng để thu thập thông tin</p> <p>Để tạo được 1 ô nhập dữ liệu cần có 1 nhãn label (có mã for bằng tên id của input) và 1 nhãn input</p> <p>First name: <input type="text" value="Nhập ở đây nè"/></p> <p>Last name: <input type="text" value="Nhập ở đây nữa"/></p> <p>2. Ví dụ cho dữ liệu ô dạng radio</p> <p><input type="radio"/> check vào đây được nè <input type="radio"/> đây nữa <input checked="" type="radio"/> nhưng dạng radio thì chỉ check vào được 1 ô thôi nha</p> <p>3. Ví dụ cho dữ liệu ô dạng checkbox - có thể chọn nhiều tick</p> <p><input checked="" type="checkbox"/> có thể check vào đây nha <input checked="" type="checkbox"/> trong lúc đó có thể chọn cả 2 luôn <input checked="" type="checkbox"/> tham thêm tí thì chọn cả 3 luôn</p> <p>4. Định dạng submit sẽ thực hiện action đã được khai báo</p> <p>First name: <input type="text" value="John"/></p> <p>Last name: <input type="text" value="Doe"/></p> <p><input type="button" value="Bấm vào đây để thực hiện hành động"/></p> <p>5. Chọn sử dụng dữ liệu có sẵn - sử dụng for và select có id giống với for</p> <p>Chọn 1 nè: <input type="text" value="Luhe"/></p>

		<p>6. Cài thêm thuộc tính multiple để có thể chọn nhiều giá trị</p>  <p>Nhập nhiều dòng vào đây được nè.</p> <p>8. Gom các phần tử quan trọng lại</p> <p>Phần tử fieldset được sử dụng để nhóm dữ liệu liên quan trong một biểu mẫu. Phần tử legend xác định chủ thích cho phần tử fieldset.</p> <p>Tiêu đề chính sử dụng nhãn legend nè:</p> <p>First name: John Last name: Doe</p> <p>9. Sử dụng nhãn datalist để thể hiện list</p>  <p>Nó sẽ hiển thị 1 hộp thoại để mở file Choose File No file chosen</p> <p>10. Sử dụng input với 2 dạng range và number</p> <p>0 <input type="range"/> 100 + <input type="text" value="50"/> = <input type="button" value="Submit"/></p> <p>11. tải file lên</p> <p>Nó sẽ hiển thị 1 hộp thoại để mở file Choose File No file chosen <input type="button" value="Submit"/></p> <p>12. Một số thuộc tính thêm của Form</p> <ul style="list-style-type: none"> - Thuộc tính autocomplete sẽ cho khả năng trình duyệt tự dự đoán diễn dưa trên các thông tin trước của người dùng - Thuộc tính readonly không cho người dùng khả năng chỉnh sửa nhưng có thể tích vào để copy - Thuộc tính disabled dấu vào chỉ định rằng trường đầu vào sẽ bị vô hiệu hóa. Trường nhập bị tắt sẽ không sử dụng được và không thể nhấp vào được. Giá trị của trường nhập bị tắt sẽ không được gửi khi submit biểu mẫu!
21. Graphics - Canvas	Code	<pre><!DOCTYPE html> <html> <body> <p>Phần tử canvas HTML được sử dụng để vẽ đồ họa một cách nhanh chóng thông qua JavaScript.

</pre> <p>Phần tử canvas chỉ là nơi chứa đồ họa. Bạn phải sử dụng JavaScript để thực sự vẽ đồ họa.</p>

```

<br><br>
    Canvas có một số phương pháp để vẽ đường dẫn, hộp, hình tròn, văn
    bản và thêm hình ảnh.</p>
<h1>1. Ví dụ đơn giản cho canvas - tạo đường line</h1>
<canvas id="myCanvas" width="200" height="100" style="border:1px
solid #000000;">
</canvas>
<h1>2. Canvas với java</h1>
<canvas id="vidu2" width="200" height="100" style="border:1px solid
#d3d3d3;"></canvas>

<script>
var c = document.getElementById("vidu2");
var ctx = c.getContext("2d");
ctx.moveTo(0,0);
ctx.lineTo(200,100);
ctx.stroke();
</script>

<h1>3. Tạo hình tròn</h1>
<canvas id="hinhtron" width="200" height="100" style="border:1px solid
#d3d3d3;">
    Your browser does not support the HTML canvas tag.</canvas>

<script>
var c = document.getElementById("hinhtron");
var ctx = c.getContext("2d");
ctx.beginPath();
ctx.arc(95,50,40,0,2*Math.PI);
ctx.stroke();
</script>

<h1>4. Tạo chữ</h1>
<canvas id="text" width="200" height="100" style="border:1px solid
#d3d3d3;">
    Your browser does not support the HTML canvas tag.</canvas>

<script>
var c = document.getElementById("text");
var ctx = c.getContext("2d");
ctx.font = "30px Arial";
ctx.fillText("Hello World",10,50);
</script>

<h1>5. Tạo chữ nhưng có thể stroke</h1>

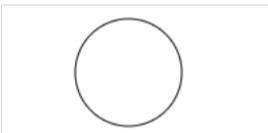
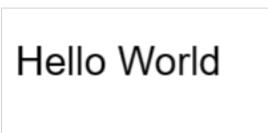
```

```
<canvas id="text2" width="200" height="100" style="border:1px solid #d3d3d3;">
    Your browser does not support the HTML canvas tag.</canvas>

<script>
var c = document.getElementById("text2");
var ctx = c.getContext("2d");
ctx.font = "30px Arial";
ctx.strokeText("Hello World",10,50);
</script>

<h1>6. Đỗ màu Gadiant</h1>
<canvas id="gadiant" width="200" height="100" style="border:1px solid #d3d3d3;">
    Your browser does not support the HTML canvas tag.</canvas>

<script>
var c = document.getElementById("gadiant");
var ctx = c.getContext("2d");
// Create gradient
var grd = ctx.createLinearGradient(0,0,200,0);
grd.addColorStop(0,"pink");
grd.addColorStop(1,"white");
// Fill with gradient
ctx.fillStyle = grd;
ctx.fillRect(10,10,150,80);
</script>
</body>
</html>
```

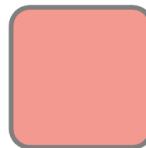
	Result	<p>Phần tử canvas HTML được sử dụng để vẽ đồ họa một cách nhanh chóng thông qua JavaScript.</p> <p>Phần tử canvas chỉ là nơi chứa đồ họa. Bạn phải sử dụng JavaScript để thực sự vẽ đồ họa.</p> <p>Canvas có một số phương pháp để vẽ đường dẫn, hộp, hình tròn, văn bản và thêm hình ảnh.</p> <p>1. Ví dụ đơn giản cho canvas - tạo đường line</p>  <p>2. Canvas với java</p>  <p>3. Tạo hình tròn</p>  <p>4. Tạo chữ</p>  <p>5. Tạo chữ nhưng có thể stroke</p>  <p>6. Đỗ màu Gadiant</p> 
21. Graphics - SVG	Code	<pre><!DOCTYPE html> <html> <body></pre>

```

<h1>1. Tạo hình tròn có đố màu với các thuộc tính width và height là khung của svg</h1>
<p>Dùng lệnh circle để tạo hình tròn</p>
<svg width="100" height="100">
  <circle cx="50" cy="50" r="40"
    stroke="pink" stroke-width="4" fill="powderblue" />
Sorry, your browser does not support inline SVG.
</svg>

<h1>2. Hình chữ nhật</h1>
<p>Dùng lệnh rect để tạo hình chữ nhật</p>
<svg width="400" height="100">
  <rect width="400" height="100"
    style="fill:rgb(0,0,255);stroke-width:10;stroke:rgb(0,0,0)" />
Sorry, your browser does not support inline SVG.
</svg>
<h1>3. Hình chữ nhật bo góc - sử dụng thêm rx và ry để tạo độ bo góc</h1>
<svg width="400" height="180">
  <rect x="50" y="20" rx="20" ry="20" width="150" height="150"
    style="fill:red;stroke:black;stroke-width:5;opacity:0.5" />
Sorry, your browser does not support inline SVG.
</svg>
<h1>4. vẽ polygon </h1>
<svg width="300" height="200">
  <polygon points="100,10 40,198 190,78 10,78 160,198"
    style="fill:lime;stroke:purple;stroke-width:5;fill-rule:evenodd;" />
Sorry, your browser does not support inline SVG.
</svg>
</body>
</html>

```

	Result	<p>1. Tạo hình tròn có đỗ màu với các thuộc tính width và height là khung của svg</p> <p>Dùng lệnh circle để tạo hình tròn</p>  <p>2. Hình chữ nhật</p> <p>Dùng lệnh rect để tạo hình chữ nhật</p>  <p>3. Hình chữ nhật bo góc - sử dụng thêm rx và ry để tạo độ bo góc</p>  <p>4. vẽ polygon</p> 
22. Video	Code	<pre><!DOCTYPE html> <html> <body> <h1>1.Thêm video vào trang bằng thẻ video, src là link video, type là định dạng của video</h1> <video width="400" controls> <source src="mov_bbb.mp4" type="video/mp4"> <source src="mov_bbb.ogg" type="video/ogg"> </video> <h1>Chúng ta cũng có thể thêm chức năng autoplay</h1> <video width="320" height="240" autoplay> <source src="movie.mp4" type="video/mp4"> <source src="movie.ogg" type="video/ogg"> Your browser does not support the video tag. </video> <h1>2. Tạo thêm các nút nhấn để điều khiển video</h1> <div style="text-align:center"> <button onclick="playPause()">Play/Pause</button> <button onclick="makeBig()">Big</button> <button onclick="makeSmall()">Small</button></pre>

```
<button onclick="makeNormal()">Normal</button>
<br><br>
<video id="video2" width="420">
  <source src="mov_bbb.mp4" type="video/mp4">
  <source src="mov_bbb.ogg" type="video/ogg">
    Your browser does not support HTML video.
</video>
</div>

<script>
var myVideo = document.getElementById("video2");

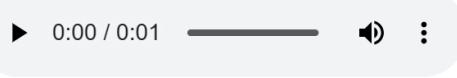
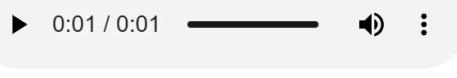
function playPause() {
  if (myVideo.paused)
    myVideo.play();
  else
    myVideo.pause();
}

function makeBig() {
  myVideo.width = 560;
}

function makeSmall() {
  myVideo.width = 320;
}

function makeNormal() {
  myVideo.width = 420;
}
</script>

</div>
</body>
</html>
```

	Result	<p>1.Thêm video vào trang bằng thẻ video, src là link video, type là định dạng của video</p>  <p>Chúng ta cũng có thể thêm chức năng autoplay</p>  <p>1. Sử dụng nhãn Audio để chèn file âm nhạc vào web</p>  <p>2. Sử dụng nhãn Audio để chèn file âm nhạc vào web - có sử dụng đặc trưng controls autoplay</p> 
23. Audio	Code	<!DOCTYPE html> <html>

		<pre> <body> <h1>1. Sử dụng nhãn Audio để chèn file âm nhạc vào web</h1> <audio controls> <source src="horse.ogg" type="audio/ogg"> <source src="horse.mp3" type="audio/mpeg"> </audio> <h1>2. Sử dụng nhãn Audio để chèn file âm nhạc vào web - có sử dụng đặc trưng controls autoplay </h1> <audio controls autoplay> <source src="horse.ogg" type="audio/ogg"> <source src="horse.mp3" type="audio/mpeg"> Your browser does not support the audio element. </audio> </body> </html> </pre>
	Result	
24. Plug-in	Code	<pre> <!DOCTYPE html> <html> <body> <!-- Plug-in được thiết kế để sử dụng cho nhiều mục đích khác nhau: Để chạy các ứng dụng Java Để chạy các điều khiển Microsoft ActiveX Để hiển thị phim Flash Để hiển thị bản đồ Để quét virus Để xác minh id ngân hàng --> <h1> 1. Ví dụ lấy thông tin trong bảng snippet.html bằng thẻ object </h1> <object width="100%" height="500px" data="snippet.html"></object> <h1> 2. Lấy hình ảnh bằng object qua thuộc tính data </h1> <object data="audi.jpeg"></object> <h1> 3. Chúng ta cũng có thể nhúng bằng thẻ embed tuy nhiên embed thường được nhúng PDF </h1> <embed src="audi.jpeg"> </body> </html> </pre>

	Result	<p>1. Ví dụ lấy thông tin trong bảng snippet.html bằng thẻ object</p> <table border="0"> <tbody> <tr><td>Alfreds Futterkiste</td><td>Berlin</td><td>Germany</td></tr> <tr><td>Berglunds snabbköp</td><td>Luleå</td><td>Sweden</td></tr> <tr><td>Centro comercial Moctezuma</td><td>México D.F.</td><td>Mexico</td></tr> <tr><td>Ernst Handel</td><td>Graz</td><td>Austria</td></tr> <tr><td>FISSA Fabrica Inter. Salchichas S.A.</td><td>Madrid</td><td>Spain</td></tr> <tr><td>Galeria del gastrónomo</td><td>Barcelona</td><td>Spain</td></tr> <tr><td>Island Trading</td><td>Cowes</td><td>UK</td></tr> <tr><td>Königlich Essen</td><td>Brandenburg</td><td>Germany</td></tr> <tr><td>Laughing Bacchus Wine Cellars</td><td>Vancouver</td><td>Canada</td></tr> <tr><td>Magazzini Alimentari Riuniti</td><td>Bergamo</td><td>Italy</td></tr> <tr><td>North/South</td><td>London</td><td>UK</td></tr> <tr><td>Paris spécialités</td><td>Paris</td><td>France</td></tr> <tr><td>Rattlesnake Canyon Grocery</td><td>Albuquerque</td><td>USA</td></tr> <tr><td>Simons bistro</td><td>København</td><td>Denmark</td></tr> <tr><td>The Big Cheese</td><td>Portland</td><td>USA</td></tr> <tr><td>Vaffeljernet</td><td>Århus</td><td>Denmark</td></tr> <tr><td>Wolski Zajazd</td><td>Warszawa</td><td>Poland</td></tr> </tbody> </table> <p>2. Lấy hình ảnh bằng object qua thuộc tính data</p>  <p>3. Chúng ta cũng có thể nhúng bảng snippet.html bằng thẻ embed tuy nhiên embed thường được nhúng PDF</p> 	Alfreds Futterkiste	Berlin	Germany	Berglunds snabbköp	Luleå	Sweden	Centro comercial Moctezuma	México D.F.	Mexico	Ernst Handel	Graz	Austria	FISSA Fabrica Inter. Salchichas S.A.	Madrid	Spain	Galeria del gastrónomo	Barcelona	Spain	Island Trading	Cowes	UK	Königlich Essen	Brandenburg	Germany	Laughing Bacchus Wine Cellars	Vancouver	Canada	Magazzini Alimentari Riuniti	Bergamo	Italy	North/South	London	UK	Paris spécialités	Paris	France	Rattlesnake Canyon Grocery	Albuquerque	USA	Simons bistro	København	Denmark	The Big Cheese	Portland	USA	Vaffeljernet	Århus	Denmark	Wolski Zajazd	Warszawa	Poland
Alfreds Futterkiste	Berlin	Germany																																																			
Berglunds snabbköp	Luleå	Sweden																																																			
Centro comercial Moctezuma	México D.F.	Mexico																																																			
Ernst Handel	Graz	Austria																																																			
FISSA Fabrica Inter. Salchichas S.A.	Madrid	Spain																																																			
Galeria del gastrónomo	Barcelona	Spain																																																			
Island Trading	Cowes	UK																																																			
Königlich Essen	Brandenburg	Germany																																																			
Laughing Bacchus Wine Cellars	Vancouver	Canada																																																			
Magazzini Alimentari Riuniti	Bergamo	Italy																																																			
North/South	London	UK																																																			
Paris spécialités	Paris	France																																																			
Rattlesnake Canyon Grocery	Albuquerque	USA																																																			
Simons bistro	København	Denmark																																																			
The Big Cheese	Portland	USA																																																			
Vaffeljernet	Århus	Denmark																																																			
Wolski Zajazd	Warszawa	Poland																																																			
25. YouTube	Code	<!DOCTYPE html> <html>																																																			

```
<body>

<h1>Sử dụng iframe và src để hiển thị youtube vào web </h1>
<iframe width="420" height="345"
src="https://www.youtube.com/embed/tgbNymZ7vqY">
</iframe>
<h1>2. Có thêm phần autoplay để mở chức năng tự chạy khi mở trình duyệt</h1>
<iframe width="420" height="345"
src="https://www.youtube.com/embed/tgbNymZ7vqY?autoplay=1&mute=1">
</iframe>

<h1>3. Cũng là thêm video mà sử dụng thêm loop - vòng lặp 1 lần</h1>
<iframe width="420" height="345"
src="https://www.youtube.com/embed/tgbNymZ7vqY&loop=1">
</iframe>

<h1>4. Thêm control=0 để không hiển thị các điều khiển trong trình phát video. Giá trị 0: Điều khiển trình phát không hiển thị. Giá trị 1 (mặc định): Hiển thị điều khiển trình phát.</h1>
<iframe width="420" height="345"
src="https://www.youtube.com/embed/tgbNymZ7vqY?controls=0">
</iframe>
</body>
</html>
```

	<p>Result</p> <p>Sử dụng iframe và src để hiển thị youtube vào web</p> 
	<p>2. Có thêm phần autoplay để mở chức năng tự chạy khi mở trình duyệt</p> 

		<p>3. Cũng là thêm video mà sử dụng thêm loop - vòng lặp 1 lần</p> 
		<p>4. Thêm control=0 để không hiển thị các điều khiển trong trình phát video. Giá trị 0: Điều khiển trình phát không hiển thị. Giá trị 1 (mặc định): Hiển thị điều khiển trình phát.</p> 
26. Geolocation	Code	<pre><!DOCTYPE html> <html> <body> <p>The HTML Geolocation API is used to locate a user's position.</p> <h1>Phương thức getCurrentPosition() được sử dụng để trả về vị trí của người dùng. Ví dụ bên dưới trả về vĩ độ và kinh độ vị trí của người dùng:</h1> <button onclick="getLocation()">Bấm vào đây để xuất vị trí nè</button> <p id="demo"></p> <script> var x = document.getElementById("demo"); function getLocation() { if (navigator.geolocation) {</pre>

		<pre> navigator.geolocation.getCurrentPosition(showPosition); } else { x.innerHTML = "Geolocation is not supported by this browser."; } } function showPosition(position) { x.innerHTML = "Nó sẽ trả về kinh độ nè: " + position.coords.latitude + "
Vĩ độ thì ở đây : " + position.coords.longitude; } </script> <p> watchPosition() - Trả về vị trí hiện tại của người dùng và tiếp tục trả về vị trí cập nhật khi người dùng di chuyển (như GPS trong xe hơi). clearWatch() - Dừng phương pháp.watchPosition()</p> </body> </html> </pre> <p>watchPosition() - Trả về vị trí hiện tại của người dùng và tiếp tục trả về vị trí cập nhật khi người dùng di chuyển (như GPS trong xe hơi). clearWatch() - Dừng phương pháp.watchPosition()</p>
Result		<p>The HTML Geolocation API is used to locate a user's position.</p> <p>Phương thức getCurrentPosition() được sử dụng để trả về vị trí của người dùng. Ví dụ bên dưới trả về vĩ độ và kinh độ vị trí của người dùng:</p> <p style="border: 1px solid black; padding: 2px; display: inline-block;">Bấm vào đây để xuất vị trí nè</p> <p>Nó sẽ trả về kinh độ nè: 10.83523 Vĩ độ thì ở đây : 106.776047</p> <p>watchPosition() - Trả về vị trí hiện tại của người dùng và tiếp tục trả về vị trí cập nhật khi người dùng di chuyển (như GPS trong xe hơi). clearWatch() - Dừng phương pháp.watchPosition()</p>

1.2 CSS

1. Color	Code	<pre> <!DOCTYPE html> <html> <body> <h1> 1. Định dạng bằng tên gọi của các màu</h1> <h1 style="background-color:pink;">Định dạng màu nền là màu hồng bằng Css qua tham chiếu style background-color nè </h1> <p style="background-color:Tomato;"> </pre>
----------	------	---

	<p>Cái này cũng vậy nhưng màu cà chua nha</p> <pre></p></pre> <p><h1 style="color:Tomato;">Cái này định dạng màu chữ nha</h1></p> <p><p style="color:DodgerBlue;">Cái này cũng là màu chữ nhưng là màu xanh nha</p></p> <p><h1 style="border: 2px solid DodgerBlue;">Định dạng khung màu xanh bằng thẻ border nè</h1></p> <p><h1 style="border: 20px solid Violet;"> Cũng là khung nhưng màu khác và to hơn nè</h1></p> <p><p>Ngoài các kiểu gọi tên màu bằng tên chúng ta còn có thể gọi tên bằng mã HEX, RGB, HSL</p></p> <pre></body> </html></pre>
Result	<p>1. Định dạng bằng tên gọi của các màu</p> <p>Định dạng màu nền là mày hồng bằng Css qua tham chiếu style background-color nè</p> <p>Cái này cũng vậy nhưng màu cà chua nha</p> <p>Cái này định dạng màu chữ nha</p> <p>Cái này cũng là màu chữ nhưng là màu xanh nha</p> <p>Định dạng khung màu xanh bằng thẻ border nè</p> <p>Cũng là khung nhưng màu khác và to hơn nè</p> <p>Ngoài các kiểu gọi tên màu bằng tên chúng ta còn có thể gọi tên bằng mã HEX, RGB, HSL</p>
Code	Background Color:

2. Background

```
<!DOCTYPE html>
<html>
<head>
<style>

h1 {
    background-color: palegreen;
}

div {
    background-color: lightblue;
    opacity: 0.6;
}

p {
    background-color: NavajoWhite;
}
</style>
</head>
<body>

<h1>Ví dụ về màu nền!</h1>
<div>
This is a text inside a div element.
<p>This paragraph has its own background color.</p>
We are still in the div element.
</div>

</body>
</html>
```

Background Image:

```
<!DOCTYPE html>
<html>
<head>
<style>
body {
    background-image: url("paper.gif");
}
</style>
</head>
<body>

<h1>Hello World!</h1>

<p>This page has an image as the background!</p>

</body>
</html>
```

Background Repeat:

		<pre> <!DOCTYPE html> <html> <head> <style> body { background-image: url("img_tree.png"); background-repeat: no-repeat; background-position: right top; margin-right: 200px; } </style> </head> <body> <h1>Hello World!</h1> <p>Here, the background image is only shown once. In addition it is positioned away from the text.</p> <p>In this example we have also added a margin on the right side, so that the background image will not disturb the text.</p> </body> </html> </pre>
Result	Background Color:	<p>Ví dụ về màu nền!</p> <p>This is a text inside a div element.</p> <p>This paragraph has its own background color.</p> <p>We are still in the div element.</p>
	Background Image:	<p>Hello World!</p> <p>This page has an image as the background!</p> <p>Background Repeat:</p> <p>Hello World!</p> <p>Here, the background image is only shown once. In addition it is positioned away from the text.</p> <p>In this example we have also added a margin on the right side, so that the background image will not disturb the text.</p> 
3. Border	Code	<pre> <!DOCTYPE html> <html> </pre>

```
<head>
    <!--Định dạng theo class-->
<style>
p.dotted {border-style: dotted;}
p.dashed {border-style: dashed;}
p.solid {border-style: solid;}
p.double {border-style: double;}
p.groove {border-style: groove;}
p.ridge {border-style: ridge;}
p.inset {border-style: inset;}
p.outset {border-style: outset;}
p.none {border-style: none;}
p.hidden {border-style: hidden;}
p.mix {border-style: dotted dashed solid double;}
p.one {
    border-style: solid;
    border-width: 5px;
    border-color: green;
}

p.two {
    border-style: solid;
    border-width: medium;
}

p.three {
    border-style: dotted;
    border-width: 2px;
}

p.four {
    border-style: dotted;
    border-width: thick;
}

p.five {
    border-style: double;
    border-width: 15px;
}

p.six {
    border-style: double;
    border-width: thick;
}
p.seven {
```

```

border-color: blue;
border-style: solid;
border-width: 25px 10px 4px 35px; /* 25px top, 10px right, 4px bottom and
35px left */
}
p.custom {
    border-top-style: dotted;
    border-right-style: solid;
    border-bottom-style: dotted;
    border-left-style: solid;
}
</style>
</head>
<body>
```

<h2>Đây là cách tạo border cho khung của thẻ 'p'</h2>

<p class="dotted">1. Dotted là định dạng khung chấm nhỏ.</p>
<p class="dashed">2. dashed là định dạng khung dạng gạch đứt.</p>
<p class="solid">3. solid là dạng khung liền mày đen</p>
<p class="double">4. double là dạng khung có 2 dòng kẻ mảnh</p>
<p class="groove">5. groove là dạng khung mảng.</p>
<p class="ridge">6. ridge là dạng khung mảng nhất.</p>
<p class="inset">7. inset là dạng khung có thanh trên và đứng trái đậm hơn 2 thanh còn lại.</p>
<p class="outset">8. outset là dạng khung có thanh dưới và đứng phải đậm hơn.</p>
<p class="none">9. None là dạng không khung.</p>
<p class="hidden">10. A hidden border.</p>
<p class="mix">11. mix là định dạng khung ẩn.</p>

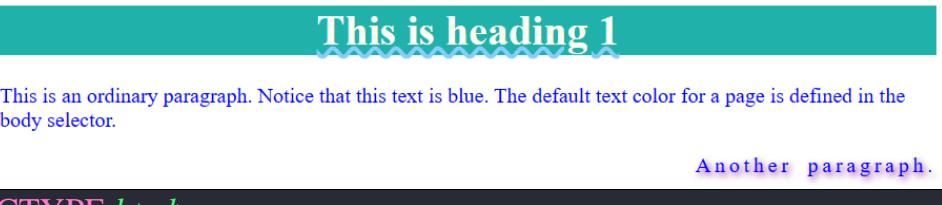
<p class="one">12. Định dạng khung dày 5px và dạng solid và có thêm màu của khung nè nè.</p>
<p class="two">13. Cũng là solid mà thêm độ dày là medium nè.</p>
<p class="three">14. dạng dotted và dày 2px nè.</p>
<p class="four">15. Cũng là định dạng dotted và có mà nó dày hơn dạng thick.</p>
<p class="five">16. Dạng double và dày 15px.</p>
<p class="six">17. Dạng double và độ dày là thick nè.</p>
<p class="seven">18. Dạng solid và độ dày được định dạng custom là 25px top, 10px right, 4px bottom and 35px left và có thêm màu của khung.</p>
<p class="custom">19. Dạng custom cho định dạng của khung nè.</p>
</body>
</html>

	Result	<p>Đây là cách tạo border cho khung của thẻ 'p'</p> <p>1. Dotted là định dạng khung chấm nhò nè.</p> <p>2. dashed là định dạng khung dạng gạch dứt.</p> <p>3. solid là dạng khung liền màu đen</p> <p>4. double là dạng khung có 2 dòng kẻ mảnh</p> <p>5. groove là dạng khung mảng.</p> <p>6. ridge là dạng khung mảng nhát.</p> <p>7. inset là dạng khung có thanh trên và đứng trái đậm hơn 2 thanh còn lại.</p> <p>8. outset là dạng khung có thanh dưới và đứng phải đậm hơn.</p> <p>9. None là dạng không khung.</p> <p>10. A hidden border.</p> <p>11. mix là định dạng khung ẩn.</p> <p>12. Định dạng khung đầy 5px và dạng solid và có thêm màu của khung nè nè.</p> <p>13. Cũng là solid mà thêm độ dày là medium nè.</p> <p>14. dạng dotted và dày 2px nè.</p> <p>15. Cũng là định dạng dotted và có mà nó dày hơn dạng thick.</p> <p>16. Dạng double và dày 15px.</p> <p>17. Dạng double và độ dày là thick nè.</p> <p>18. Dạng solid và độ dày được định dạng custom là 25px top, 10px right, 4px bottom and 35px left và có thêm màu của khung.</p> <p>19. Dạng custom cho định dạng của khung nè.</p>
4. Margin	Code	<pre><!DOCTYPE html> <html> <head> <style> div { border: 2px solid black; margin-top: 80px; margin-bottom: 100px; margin-right: 150px; margin-left: 80px; background-color: lightblue; } </style> </head> <body> <h2>Using individual margin properties</h2> <div>This div element has a top margin of 100px, a right margin of 150px, a bottom margin of 100px, and a left margin of 80px.</div> </body> </html></pre>

	Result	Using individual margin properties <div style="border: 1px solid black; padding: 5px; background-color: #e0f2f1;">This div element has a top margin of 100px, a right margin of 150px, a bottom margin of 100px, and a left margin of 80px.</div>
5. Padding	Code	<pre><!DOCTYPE html> <html> <head> <style> div.one { border: 1px solid black; padding: 25px 50px 75px 100px; background-color: pink; } div.ex1 { width: 300px; background-color: yellow; } div.ex2 { width: 300px; padding: 25px; box-sizing: border-box; background-color: lightblue; } </style> </head> <body> <h2>CSS Padding</h2> <div class="one">Một ví dụ cho phần padding dạng solid đen và có màu hồng với định dạng 25px 50px 75px 100px .</div> <div class="ex1">Định dạng cho chiều ngang là 300px nè.</div>
 <div class="ex2">Chiều rộng của div này vẫn ở mức 300px, mặc dù tổng số phần đệm bên trái và bên phải là 50px, do thuộc tính box-sizing: border-box.</div> </body> </html></pre>

	Result	<h2>CSS Padding</h2> <div style="background-color: #f08080; padding: 10px;"> <p>Một ví dụ cho phần padding dạng solid đen và có màu hồng với định dạng 25px 50px 75px 100px .</p> </div> <div style="background-color: #d0e0ff; padding: 10px;"> <p>Định dạng cho chiều ngang là 300px nè.</p> </div>
6. Height	Code	<pre><!DOCTYPE html> <html> <head> <style> div { max-width: 500px; height: 50px; background-color: powderblue; } </style> </head> <body> <h2>Set the max-width of an element</h2> <div>This div element has a height of 100px and a max-width of 500px.</div> <p>Resize the browser window to see the effect.</p> </body> </html></pre>
	Result	<p>Set the max-width of an element</p> <div style="background-color: #d0e0ff; padding: 10px;"> <p>This div element has a height of 100px and a max-width of 500px.</p> </div> <p>Resize the browser window to see the effect.</p>
7. Outline	Code	<pre><!DOCTYPE html> <html> <head> <style> p {outline-color:red;} p.dotted {outline-style: dotted;} p.dashed {outline-style: dashed;} p.solid {outline-style: solid;} p.double {outline-style: double;}</pre>

	<pre> <p><i>p.groove</i> {outline-style: groove;}</p> <p><i>p.ridge</i> {outline-style: ridge;}</p> <p><i>p.inset</i> {outline-style: inset;}</p> <p><i>p.outset</i> {outline-style: outset;}</p> </style> </head> <body> <h2>The outline-style Property</h2> <p <i>class</i>="dotted">1. Dotted là định dạng khung chấm nhỏ nè.</p> <p <i>class</i>="dashed">2. dashed là định dạng khung dạng gạch đứt.</p> <p <i>class</i>="solid">3. solid là dạng khung liền mày đen</p> <p <i>class</i>="double">4. double là dạng khung có 2 dòng kẻ mảnh</p> <p <i>class</i>="groove">5. groove là dạng khung mảng.</p> <p <i>class</i>="ridge">6. ridge là dạng khung mảng nhất.</p> <p <i>class</i>="inset">7. inset là dạng khung có thanh trên và đứng trái đậm hơn 2 thanh còn lại.</p> <p <i>class</i>="outset">8. outset là dạng khung có thanh dưới và đứng phải đậm hơn.</p> </body> </html> </pre>
Result	<h3>The outline-style Property</h3> <p>1. Dotted là định dạng khung chấm nhỏ nè.</p> <p>2. dashed là định dạng khung dạng gạch đứt.</p> <p>3. solid là dạng khung liền mày đen</p> <p>4. double là dạng khung có 2 dòng kẻ mảnh</p> <p>5. groove là dạng khung mảng.</p> <p>6. ridge là dạng khung mảng nhất.</p> <p>7. inset là dạng khung có thanh trên và đứng trái đậm hơn 2 thanh còn lại.</p> <p>8. outset là dạng khung có thanh dưới và đứng phải đậm hơn.</p>

8. Text	Code	<pre> <html> <head> <style> body { color: blue; } h1 { color: Ivory; background-color:LightSeaGreen; text-align: center; text-decoration-line: underline; text-decoration-color: LightSkyBlue; text-decoration-style: wavy; } p.a{ text-align: right; letter-spacing:3px; word-spacing:5px; text-shadow: 2px 2px 5px purple; } </style> </head> <body> <h1>This is heading 1</h1> <p>This is an ordinary paragraph. Notice that this text is blue. The default text color for a page is defined in the body selector.</p> <p class="a">Another paragraph.</p> </body> </html> </pre>
	Result	 <p>This is heading 1</p> <p>This is an ordinary paragraph. Notice that this text is blue. The default text color for a page is defined in the body selector.</p> <p>Another paragraph.</p>
9. Font	Code	<pre> <!DOCTYPE html> <html> <head> <style> .p1 { font-family: "Times New Roman", Times, serif; } .p2 { font-family: Arial, Helvetica, sans-serif; } .p3 { font-family: "Lucida Console", "Courier New", monospace; } </style> </head> <body> <h1>Định dạng font chữ bằng cách sử dụng font-family</h1> </pre>

		<pre><p class="p1">Định dạng font chữ Times New Roman font.</p> <p class="p2">Định dạng font chữ Arial font.</p> <p class="p3">Định dạng font chữ Lucida Console font.</p> </body> </html></pre>
	Result	<h2>Định dạng font chữ bằng cách sử dụng font-family</h2> <p>Định dạng font chữ Times New Roman font.</p> <p>Định dạng font chữ Arial font.</p> <p>Định dạng font chữ Lucida Console font.</p>
10. Icon	Code	<pre><!DOCTYPE html> <html> <head> <title>Google Icons</title> <meta name="viewport" content="width=device-width, initial-scale=1"> <link rel="stylesheet" href="https://fonts.googleapis.com/icon?family=Material+Icons"> </head> <body> <h1>Google icon library</h1> <p>Some Google icons:</p> <i class="material-icons">cloud</i> <i class="material-icons">favorite</i> <i class="material-icons">attachment</i> <i class="material-icons">computer</i> <i class="material-icons">traffic</i>

 <p>Styled Google icons (size and color):</p> <i class="material-icons" style="font-size:24px;">cloud</i> <i class="material-icons" style="font-size:36px;">cloud</i> <i class="material-icons" style="font-size:48px;color:red;">cloud</i> <i class="material-icons" style="font-size:60px;color:lightblue;">cloud</i> </body> </html></pre>
	Result	<h2>Google icon library</h2> <p>Some Google icons:</p>  <p>Styled Google icons (size and color):</p> 

11.List	Code	<pre> <!DOCTYPE html> <html> <head> <style> ol { background: #ff9999; padding: 20px; } ul { background: #3399ff; padding: 20px; } ol li { background: #ffe5e5; color: darkred; padding: 5px; margin-left: 35px; } ul li { background: #cce5ff; color: darkblue; margin: 5px; } </style> </head> <body> <h1>Định dạng bằng việc khai báo ol,ul,ol li,...</h1> Cái này ol được định dạng màu hồng đậm và padding là 20px ul thì được chọn màu nhạt hơn Cái này định dạng ul là màu xanh đậm Định dạng ul li là màu xanh nhạt hơn </body> </html> </pre>
---------	------	--

	Result	<h2>Định dạng bằng việc khai báo ol,ul,ol li,...</h2> <p>1. Cái này ol được định dạng màu hồng đậm và padding là 20px 2. ul thì được chọn màu nhạt hơn</p> <ul style="list-style-type: none"> • Cái này định dạng ul là màu xanh đậm • Định dạng ul li là màu xanh nhạt hơn
12. Table	Code	<pre><!DOCTYPE html> <html> <head> <style> table, td, th { border: 1px solid #ddd; text-align: center; } table { border-collapse: collapse; width: 100%; } th, td { padding: 15px; } </style> </head> <body> <h2>The padding Property</h2> <p>This property adds space between the border and the content in a table.</p> <table> <tr> <th>Firstname</th> <th>Lastname</th> <th>Savings</th> </tr> <tr> <td>Peter</td> <td>Griffin</td> <td>\$100</td> </tr> <tr> <td>Lois</td> <td>Griffin</td> <td>\$150</td> </tr> <tr> <td>Joe</td> <td>Swanson</td> <td>\$300</td> </tr> <tr> <td>Cleveland</td> <td>Brown</td> <td>\$250</td> </tr> </table> </body> </html></pre>

	Result	<p>The padding Property</p> <p>This property adds space between the border and the content in a table.</p> <table border="1"> <thead> <tr> <th>Firstname</th><th>Lastname</th><th>Savings</th></tr> </thead> <tbody> <tr> <td>Peter</td><td>Griffin</td><td>\$100</td></tr> <tr> <td>Lois</td><td>Griffin</td><td>\$150</td></tr> <tr> <td>Joe</td><td>Swanson</td><td>\$300</td></tr> <tr> <td>Cleveland</td><td>Brown</td><td>\$250</td></tr> </tbody> </table>	Firstname	Lastname	Savings	Peter	Griffin	\$100	Lois	Griffin	\$150	Joe	Swanson	\$300	Cleveland	Brown	\$250
Firstname	Lastname	Savings															
Peter	Griffin	\$100															
Lois	Griffin	\$150															
Joe	Swanson	\$300															
Cleveland	Brown	\$250															
13. Navigation	Code	<pre><!DOCTYPE html> <html> <style> ul.ex2 { list-style-type: none; margin: 0; padding: 0; width: 200px; background-color: #f1f1f1; } li a.ex2 { display: block; color: #000; padding: 8px 16px; text-decoration: none; } /* Change the link color on hover */ li a.ex2:hover { background-color: #555; color: white; } ul.ex3 { list-style-type: none; margin: 0; padding: 0; overflow: hidden; background-color: #333; } li.ex3 { float: left; } li a.ex3 { display: block;</pre>															

```

        color: white;
        text-align: center;
        padding: 14px 16px;
        text-decoration: none;
    }

    li a.ex3:hover {
        background-color: #111;
    }

```

</style>

<body>

<h1>Navigation giúp tạo các thẻ giúp trang web đỡ nhảm chán</h1>

<p>Ví dụ cơ bản cho việc đặt thẻ là thêm # vào trước tên của thuộc tính href</p>

- Home
- News
- Contact
- About

<p> 2. Ví dụ 2 kèm thêm vài định dạng và hiệu ứng</p>

<ul class="ex2">

- <li class="ex3">1. Định dạng ui với width 200px và màu nền xám
- <li class="ex">2. Thêm hành động hover khi di chuột vào sẽ hiện thành màu xám đậm hơn

<p> 3. Ví dụ cho việc chuyển các tùy chọn thành dạng ngang bằng cách thêm dòng 'float: left' vào li</p>

<ul class="ex3">

- <li class="ex3">1. Định dạng ui với width 200px và màu nền xám
- <li class="ex3">2. Thêm hành động hover khi di chuột vào sẽ hiện thành màu xám đậm hơn

</body>

</html>

	Result	<p>Navigation giúp tạo các thẻ giúp trang web đỡ nhảm chán</p> <p>Ví dụ cơ bản cho việc đặt thẻ là thêm # vào trước tên của thuộc tính href</p> <ul style="list-style-type: none"> • Home • News • Contact • About <p>2. Ví dụ 2 kèm thêm vài định dạng và hiệu ứng</p> <p>1. Định dạng ui với width 200px và màu nền xám</p> <p>2. Thêm hành động hover khi di chuột vào sẽ hiện thành màu xám đậm hơn</p> <p>3. Ví dụ cho việc chuyển các tùy chọn thành dạng ngang bằng cách thêm dòng 'float: left;'</p> <p>1. Định dạng ui với width 200px và màu nền xám 2. Thêm hành động hover khi di chuột vào sẽ hiện thành màu xám đậm hơn</p>
14. Image Gallery	Code	<pre><!DOCTYPE html> <html> <head> <style> div.gallery { margin: 5px; border: 1px solid #ccc; float: left; width: 180px; } div.gallery:hover { border: 1px solid #777; } div.gallery img { width: 100%; height: auto; } div.desc { padding: 15px; text-align: center; } </style> </head> <body> <div class="gallery"> </pre>

		<pre><div class="desc">Image Gallery giúp chúng ta có thể định dạng các hình ảnh cho nó thêm phần sinh động</div> </div> <div class="gallery"> <div class="desc">Nó sử dụng thẻ div và class gallery</div> </div> <div class="gallery"> <div class="desc">Với các chức năng đang được ứng dụng là định dạng, hover </div> </div> </body> </html></pre>
	Result	<div style="display: flex; justify-content: space-around;">    </div> <div style="display: flex; justify-content: space-around; margin-top: 10px;"> <div style="text-align: center;">  <p>Image Gallery giúp chúng ta có thể định dạng các hình ảnh cho nó thêm phần sinh động</p> </div> <div style="text-align: center;">  <p>Nó sử dụng thẻ div và class gallery</p> </div> <div style="text-align: center;">  <p>Với các chức năng đang được ứng dụng là định dạng, hover</p> </div> </div>
15. Website Layout	Code	<pre><!DOCTYPE html> <html> <head> <style> * { box-sizing: border-box; } body { font-family: Arial; padding: 10px; } </pre>

```
background: #f1f1f1;
}

/* Header/Blog Title */
.header {
    padding: 30px;
    text-align: center;
    background: white;
}

.header h1 {
    font-size: 50px;
}

/* Style the top navigation bar */
.topnav {
    overflow: hidden;
    background-color: #333;
}

/* Style the topnav links */
.topnav a {
    float: left;
    display: block;
    color: #f2f2f2;
    text-align: center;
    padding: 14px 16px;
    text-decoration: none;
}

/* Change color on hover */
.topnav a:hover {
    background-color: #ddd;
    color: black;
}

/* Create two unequal columns that floats next to each other */
/* Left column */
.leftcolumn {
    float: left;
    width: 75%;
}

/* Right column */
.rightcolumn {
```

```

        float: left;
        width: 25%;
        background-color: #f1f1f1;
        padding-left: 20px;
    }

/* Fake image */
fakeimg {
    background-color: #aaa;
    width: 100%;
    padding: 20px;
}

/* Add a card effect for articles */
.card {
    background-color: white;
    padding: 20px;
    margin-top: 20px;
}

/* Clear floats after the columns */
.row::after {
    content: "";
    display: table;
    clear: both;
}

/* Footer */
footer {
    padding: 20px;
    text-align: center;
    background: #ddd;
    margin-top: 20px;
}

/* Responsive layout - when the screen is less than 800px wide, make the
two columns stack on top of each other instead of next to each other */
@media screen and (max-width: 800px) {
    .leftcolumn, .rightcolumn {
        width: 100%;
        padding: 0;
    }
}

```

```

/* Responsive layout - when the screen is less than 400px wide, make the
navigation links stack on top of each other instead of next to each other */
@media screen and (max-width: 400px) {
    .topnav a {
        float: none;
        width: 100%;
    }
}
</style>
</head>
<body>

<div class="header">
    <h1>Thẻ h1 trong header</h1>
    <p>Thẻ p trong header</p>
</div>

<div class="topnav">
    <a href="#">Thanh nav</a>
    <a href="#">Sử dụng div và thẻ a</a>
    <a href="#">Thuộc tính href = một liên kết trong vd này là #</a>
    <a href="#" style="float:right">Sử dụng style float right nên nằm bên
phải</a>
</div>

<div class="row">
    <div class="leftcolumn">
        <div class="card">
            <h2>Vào thẻ Heading 1</h2>
            <div class="fakeimg" style="height:200px;">Một hình ảnh được thêm
vào</div>
            <p>Thẻ heading 1 được định dạng với div class = leftcolumn <br>nên sẽ
được nằm phía bên trái và được ưu tiên đầu tiên</p>
            <p>Class leftcolumn làm được điều đó là nhờ vào câu lệnh @media screen
...</p>
            <div class="card">
                <h2>Thẻ Heading thứ 2</h2>
                <div class="fakeimg" style="height:200px;">Image</div>
                <p>Tương tự thẻ Heading 1 nhưng nó sẽ được ưu tiên số thứ tự thứ 2</p>
            </div>
        </div>
        <div class="rightcolumn">
            <div class="card">
                <h2>Đây là vào cột ngang phía bên phải đầu tiên</h2>
                <div class="fakeimg" style="height:100px;">Image</div>

```

```
<p>Phần div này nằm ở bên phải vì có thuộc tính của class rightcolum</p>
</div>
<div class="card">
  <h3>Hàng thứ 2 của phần rightcolum</h3>
  <div class="fakeimg"><p>Image</p></div>
  <div class="fakeimg"><p>Image</p></div>
  <div class="fakeimg"><p>Image</p></div>
</div>
<div class="card">
  <h3>Hàng thứ 3 của phần rightcolum</h3>
  <p>Some text..</p>
</div>
</div>

<div class="footer">
  <h2>Phần kết</h2>
</div>

</body>
</html>
```

	<p>Result</p> <h1>Thẻ h1 trong header</h1> <p>Thẻ p trong header</p> <p>Thanh nav Sử dụng div và thẻ a Thuộc tính href = một liên kết trong vd này là #</p> <p>Có sử dụng thêm stule float right nên nằm bên phải</p> <h2>Vào thẻ Heading 1</h2> <p>Một hình ảnh được thêm vào</p> <p>Thẻ heading 1 được định dạng với div class = leftcolumn nên sẽ được nằm phía bên trái và được ưu tiên đầu tiên</p> <p>Class leftcolumn làm được điều đó là nhờ vào câu lệnh @media screen ...</p>
--	---

Thẻ Heading thứ 2

Image

Tương tự thẻ Heading 1 nhưng nó sẽ được ưu tiên số thứ tự thứ 2

Đây là vào cột ngang phía bên phải đầu tiên

Image

Phần div này nằm ở bên phải vì có thuộc tính của class rightcolum

Hàng thứ 2 của phần rightcolum

Image

Image

Image

Hàng thứ 3 của phần rightcolum

Some text..

Phần kết

16. Rounde d Corners	Code	<pre> <!DOCTYPE html> <html> <head> <style> #rcorners1 { /*một phần tử có góc bo tròn, màu nền mặc định*/ border-radius: 25px; /*bo tròn góc*/ background: #73AD21; /*nền màu #73AD21*/ padding: 20px; /*lề*/ width: 200px; height: 150px; /*chiều ngang và dọc*/ } #rcorners2 { border-radius: 25px; border: 2px solid #73AD21; /*tạo ra khung viền nền màu #73AD21 có độ dày 2px*/ padding: 20px; width: 200px; height: 150px; } #rcorners3 { border-radius: 25px; background: url(paper.gif); /*sử dụng một hình nền paper.gif*/ background-position: left top; /*xác định vị trí ban đầu của hình ảnh*/ background-repeat: repeat; /*chỉ định hình ảnh nền được lặp lái*/ padding: 20px; width: 200px; height: 150px; } </style> </head> </pre>
-------------------------------	------	--

```
<body>

<h1>The border-radius Property</h1>

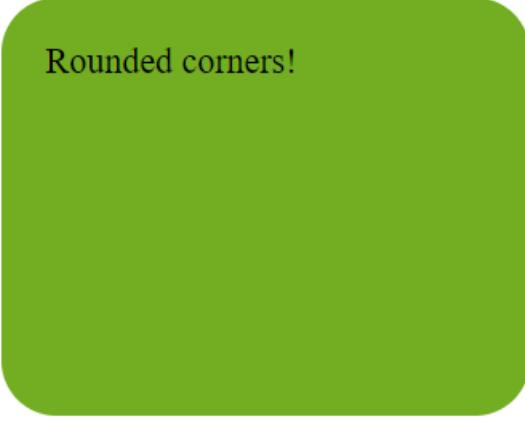
<p>Rounded corners for an element with a specified background color:</p>
<p id="rcorners1">Rounded corners!</p>
<p>Rounded corners for an element with a border:</p>
<p id="rcorners2">Rounded corners!</p>
<p>Rounded corners for an element with a background image:</p>
<p id="rcorners3">Rounded corners!</p>

</body>
</html>
```

Result

The border-radius Property

Rounded corners for an element with a specified background color:



Rounded corners!

		<p>Rounded corners for an element with a border:</p> 
17. Multiple Background	Code	<pre><!DOCTYPE html> <html> <head> <style> #example1 { background-image: url(img_flwr.gif), url(paper.gif); background-position: right bottom, left top; background-repeat: no-repeat, repeat; /*có thể thay thế ba dòng trên bằng 'background: url(img_flwr.gif) right bottom no-repeat, url(paper.gif) left top repeat; */ padding: 15px; } #example2 { border: 1px solid black; background: url(img_flwr.gif); background-size: 100px 80px; background-repeat: no-repeat; padding: 15px; /*tạo khung viền đen, sử dụng ảnh img_flwr.gif và size được báo trước*/ } #example1, #example2</pre>

```

#example3 {
    background: url(img_tree.gif) left top no-repeat, url(img_flwr.gif) right
    bottom no-repeat, url(paper.gif) left top repeat;
    padding: 15px;
    background-size: 50px, 130px, auto;
    /*có thể sử dụng nhiều ảnh gif*/
}
#example4 {
    border: 10px dotted black;
    padding: 35px;
    background: yellow;
    /*sử dụng nền màu vàng mà khung kiểu dotted*/
}

</style>
</head>
<body>

<h1>Multiple Backgrounds</h1>
<p>Phần tử này dùng 2 hình nền:</p>
<div id="example1">
    <h1>Lorem Ipsum Dolor</h1>
    <p>Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam
    nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat
    volutpat.</p>
</div>

<h1>The background-size Property</h1>
<p>thay đổi kích thước ảnh:</p>
<div id="example2">
    <h2>Lorem Ipsum Dolor</h2>
    <p>Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam
    nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat
    volutpat.</p>
</div>

<h1>Dùng 3 hình nền</h1>
<div id="example3">
    <h1>Lorem Ipsum Dolor</h1>
    <p>Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam
    nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat
    volutpat.</p>
</div>

<h1>The background-clip Property</h1>
<p>mặc định hộp viền</p>
<div id="example4">
    <h2>Lorem Ipsum Dolor</h2>
    <p>Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam
    nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat
    volutpat.</p>
</div>

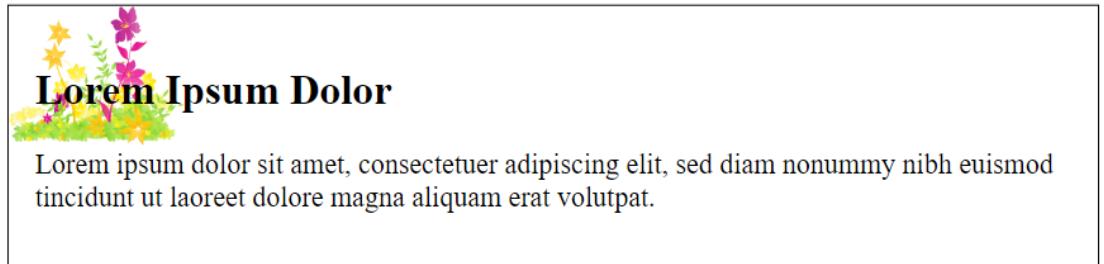
</body>
</html>

```

	Result	<h1>Multiple Backgrounds</h1> <p>Phần tử này dùng 2 hình nền:</p> <div style="background-color: #e0e0e0; padding: 10px;"><p> Lorem Ipsum Dolor</p><p> Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat.</p></div>
--	--------	---

The background-size Property

thay đổi kích thước ảnh:



Dùng 3 hình nền



The background-clip Property

mặc định hộp viền



18. Buttons	Code	<pre> <!DOCTYPE html> <html> <head> <style> .button { /*tạo một button*/ background-color: #4CAF50; border: none; color: white; padding: 15px 32px; text-align: center; text-decoration: none; display: inline-block; font-size: 16px; margin: 4px 2px; cursor: pointer } .button2 {background-color: #008CBA;} /* Blue */ /*còn nền xanh nước*/ .button3 {background-color: #e7e7e7; color: black;} /* Gray */ /*có nền màu xám, chữ màu đen*/ .button4 {font-size: 10px;} /*có kích cỡ chữ tùy ý (10px)*/ .button5 {font-size: 12px;} .button6 {padding: 10px 24px;} /*kích cỡ chữ theo width và height*/ .button7 {padding: 32px 16px;} .button8 {padding: 16px;} .button9 {border-radius: 4px;} /*tạo độ cong của góc của nút*/ .button10 {border-radius: 12px;} .button11 {border-radius: 50%;} .button12 { background-color: white; color: black; border: 2px solid #4CAF50; /*tạo khung viền cho nút với màu #4CAF50*/ } </pre>
----------------	------	---

```

.button13 {
    background-color: white;
    color: black;
    border: 2px solid #008CBA;
}
.button14 {
    background-color: white;
    color: black;
    border: 2px solid #e7e7e7;
}
.button12:hover {
    background-color: #4CAF50;
    color: white;
}
/*sử dụng hover để thay đổi kiểu dáng khi di chuyển qua nó*/
.button13:hover {
    background-color: #008CBA;
    color: white;
}
.button14:hover {background-color: #e7e7e7;}
.button15 {
    box-shadow: 0 8px 16px 0 rgba(0,0,0,0.2), 0 6px 20px 0 rgba(0,0,0,0.19);
}
/*tạo ra cái bóng xung quanh nút*/
.button16:hover {
    box-shadow: 0 12px 16px 0 rgba(0,0,0,0.24), 0 17px 50px 0 rgba(0,0,0,0.19);
}
/*tạo ra bóng khi di chuyển qua nút*/
.disabled {
    opacity: 0.6;
    cursor: not-allowed;
}
/*class disabled không cho phép nhấn vào khi di chuyển qua nút*/
.btn-group .button17 {
    background-color: #4CAF50; /* Green */
    border: none;
    color: white;
    padding: 15px 32px;
    text-align: center;
    text-decoration: none;
    display: inline-block;
    font-size: 16px;
    cursor: pointer;
    float: left;
}

```

```

.btn-group .button18:not(:last-child) {
    border-right: none; /* make các nút nằm ngang */
    /*chiều dọc sẽ là border-bottom*/
}
.btn-group .button17:hover {
    background-color: #3e8e41;
}
.button19 {
    display: inline-block;
    border-radius: 4px;
    background-color: #f4511e;
    border: none;
    color: #FFFFFF;
    text-align: center;
    font-size: 28px;
    padding: 20px;
    width: 200px;
    transition: all 0.5s;
    cursor: pointer;
    margin: 5px;
}
.button19 span {
    cursor: pointer;
    /*con trỏ chuột sẽ thành bàn tay khi qua phần tử span*/
    display: inline-block;
    position: relative;
    transition: 0.5s;
}
.button19 span:after {
    content: '\00bb';
    /*hiển thị ký tự >> làm nội dung khi di chuyển đến*/
    position: absolute; /*vị trí tương đối k ảnh hưởng đến phần tử khác*/
    opacity: 0; /*ẩn đi lúc ban đầu*/
    top: 0;
    right: -20px;
    /*đặt nằm cạnh phía bên phải của phần tử cha lệch 20pixel*/
    transition: 0.5s;
}
.button19:hover span {
    padding-right: 25px;
}
.button19:hover span:after {
    opacity: 1;
    right: 0;
}

```

```

.container {
    position: relative;
    width: 100%;
    max-width: 400px;
}

.container img {
    width: 100%;
    height: auto;
}

.container .btn {
    position: absolute;
    top: 50%;
    left: 50%;
    transform: translate(-50%, -50%);
    -ms-transform: translate(-50%, -50%);
    background-color: #f1f1f1;
    color: black;
    font-size: 16px;
    padding: 16px 30px;
    border: none;
    cursor: pointer;
    border-radius: 5px;
    text-align: center;
}

.container .btn:hover {
    background-color: black;
    color: white;
}

```

</style>

</head>

<body>

<h2>CSS Buttons</h2>

<button>Default Button</button>

Link Button

<button class="button">Button</button>

<input type="button" class="button" value="Input Button">

<h2>Button Colors</h2>

<p>Change the background color of a button with the background-color property:</p>

<button class="button">Green</button>

<button class="button button2">Blue</button>

<button class="button button3">Gray</button>

```

<h2>Button Sizes</h2>
<p>Change the font size of a button with the font-size property:</p>
<button class="button button4">10px</button>
<button class="button button5">12px</button>

<h2>Button Padding</h2>
<p>Change the padding of a button with the padding property:</p>
<button class="button button6">10px 24px</button>
<button class="button button7">32px 16px</button>
<button class="button button8">16px</button>

<h2>Rounded Buttons</h2>
<p>Add rounded corners to a button with the border-radius property:</p>
<button class="button button9">4px</button>
<button class="button button10">12px</button>
<button class="button button11">50%</button>

<h2>Colored Button Borders</h2>
<p>Use the border property to add a border to the button:</p>
<button class="button button12">Green</button>
<button class="button button13">Blue</button>
<button class="button button14">Gray</button>

<h2>Hoverable Buttons</h2>
<p>Use the :hover selector to change the style of the button when you move the mouse over it.</p>
<p><strong>Tip:</strong> Use the transition-duration property to determine the speed of the "hover" effect:</p>
<button class="button button12">Green</button>
<button class="button button13">Blue</button>
<button class="button button14">Gray</button>

<h2>Shadow Buttons</h2>
<p>Use the box-shadow property to add shadows to the button:</p>
<button class="button button15">Shadow Button</button>
<button class="button button16">Shadow on Hover</button>

<h2>Disabled Button</h2>
<p>Use the opacity property to add some transparency to a button (make it look disabled):</p>
<button class="button">Normal Button</button>
<button class="button disabled">Disabled Button</button>

```

	<pre> <h2>Button Groups</h2> <p>Remove margins and float the buttons to create a button group:</p> <div class="btn-group"> <button class="button17">Button</button> <button class="button17">Button</button> <button class="button17">Button</button> <button class="button17">Button</button> </div> <p style="clear:both">
Remember to clear floats after, or else will this p element also float next to the buttons.</p> <h2>Bordered Button Group</h2> <p>Add borders to create a bordered button group:</p> <div class="btn-group"> <button class="button18">Button</button> <button class="button18">Button</button> <button class="button18">Button</button> <button class="button18">Button</button> </div> <p style="clear:both">
Remember to clear floats after, or else will this p element also float next to the buttons.</p> <h2>Animated Button</h2> <button class="button19" style="vertical-align:middle">Hover </button> <h2>Button on Image</h2> <p>Add a button on an image:</p> <div class="container"> <button class="btn">Button</button> </div> </pre>
Result	

CSS Buttons

Default Button

Link Button

Button

Input Button

Button Colors

Change the background color of a button with the background-color property:

Green

Blue

Gray

Button Sizes

Change the font size of a button with the font-size property:

10px

12px

Button Padding

Change the padding of a button with the padding property:

10px 24px

32px 16px

16px

Rounded Buttons

Add rounded corners to a button with the border-radius property:

4px

12px

50%

Colored Button Borders

Use the border property to add a border to the button:

Green

Blue

Gray

Hoverable Buttons

Use the :hover selector to change the style of the button when you move the mouse over it.

Tip: Use the transition-duration property to determine the speed of the "hover" effect:

Green

Blue

Gray

Shadow Buttons

Use the box-shadow property to add shadows to the button:

Shadow Button

Shadow on Hover

Disabled Button

Use the opacity property to add some transparency to a button (make it look disabled):

Normal Button

Disabled Button

Button Groups

Remove margins and float the buttons to create a button group:

Button

Button

Button

Button

Remember to clear floats after, or else will this p element also float next to the buttons.

Bordered Button Group

Add borders to create a bordered button group:

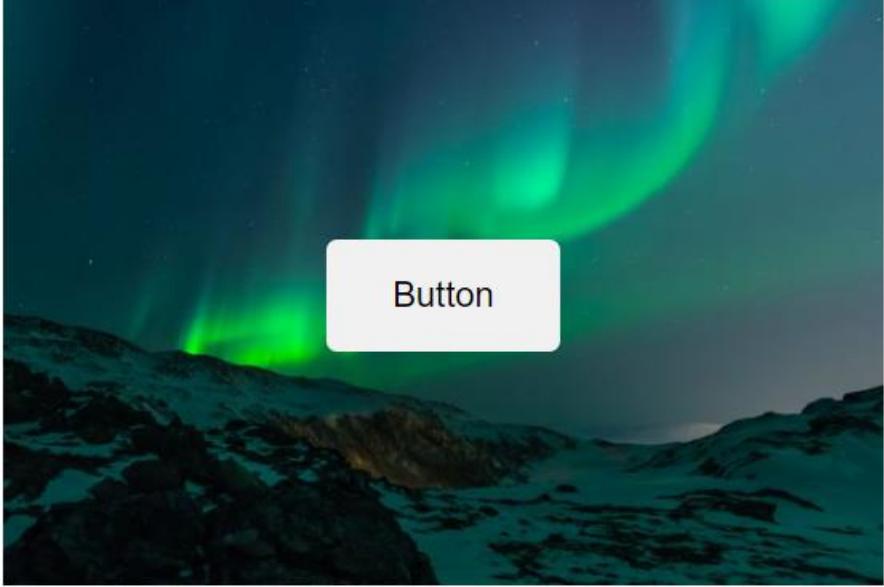
Button

Button

Button

Button

Remember to clear floats after, or else will this p element also float next to the buttons.

		<h2>Animated Button</h2>  <h2>Button on Image</h2> <p>Add a button on an image:</p> 
19. Object position	Code	<pre><!DOCTYPE html> <html> <head> <style> img { width: 200px; height: 300px; object-fit: cover; } img.ex2 { width: 200px; height: 300px; object-fit: cover; object-position: 15% 100%; } </style> </head></pre>

	<pre> <body> <h1>Thuộc tính vị trí đối tượng CSS được sử dụng để chỉ định cách định vị img hoặc video trong vùng chứa của nó.</h1> <h2>Sử dụng object-fit: cover để giữ tỷ lệ khung hình và điền vào kích thước nhất định. Tuy nhiên, hình ảnh sẽ được cắt bớt cho phù hợp</h2> <h1>Sử dụng thêm thuộc tính object-position để cắt ảnh sao cho vừa ý, giá trị đầu là chiều ngang, thứ 2 là chiều dọc.</h1> </body> </html> </pre>
Result	<p>Thuộc tính vị trí đối tượng CSS được sử dụng để chỉ định cách định vị img hoặc video trong vùng chứa của nó.</p> <p>Sử dụng object-fit: cover để giữ tỷ lệ khung hình và điền vào kích thước nhất định. Tuy nhiên, hình ảnh sẽ được cắt bớt cho phù hợp</p> 

		<p>Sử dụng thêm thuộc tính object-position để cắt ảnh sao cho vừa ý, giá trị đầu là chiều ngang, thứ 2 là chiều dọc.</p> 
20. Multiple Column s	Code	<pre> <!DOCTYPE html> <html> <head> <style> .newspaper { /*tạo hiệu ứng trang báo với nhiều cột*/ column-count: 3; /*số cột là 3*/ column-gap: 40px; /*khoảng cách giữ các cột*/ column-rule-style: solid; /*tạo ra đường viền giữa các cột*/ column-rule-width: 1px; /*độ dày đường viền*/ column-rule-color: lightblue; /*màu sắc của đường viền*/ /*column rule style, width, color có thể thay đổi bằng một lệnh ngắn: 'column-rule: 1px solid lightblue;'*/ } h1 { column-span: all; } /*title h1 được kéo dài qua tất cả các cột với giá trị all*/ </style> </head></pre>

		<pre> <body> <div class="newspaper"> <h1>Set the Rule Color</h1> <!--thẻ h1 nằm trong phần tử div nếu không có column-span ở style thì title này sẽ bị nằm trong cột đầu tiên--> Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat. Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui blandit praesent luptatum zzril delenit augue duis dolore te feugait nulla facilisi. Nam liber tempor cum soluta nobis eleifend option congue nihil imperdiet doming id quod mazim placerat facer possim assum. </div> </body> </html> </pre>
	Result	<h1>Set the Rule Color</h1> <p> Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat. Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui blandit praesent luptatum zzril delenit augue duis dolore te feugait nulla facilisi. Nam liber tempor cum soluta nobis eleifend option congue nihil imperdiet doming id quod mazim placerat facer possim assum. </p>

II. CHƯƠNG 2: TÌM HIỂU VỀ FIREBASE

1. Giới thiệu:

- Firebase là một nền tảng giúp phát triển các ứng dụng di động trong web. Bên cạnh đó, Firebase còn được hiểu là một dịch vụ cơ sở dữ liệu hoạt động trên nền tảng đám mây cloud với hệ thống máy chủ mạnh mẽ của Google.
- Firebase chứa cơ sở dữ liệu mang đến khả năng code nhanh và thuận tiện hơn. Lập trình viên có thể dễ dàng lập trình ứng dụng bằng cách đơn giản hóa các thao tác với cơ sở dữ liệu sẵn có.

2. Ưu nhược điểm:

- **Ưu điểm:**
 - + **Sử dụng miễn phí và thuận tiện:** Bạn có thể sử dụng những tính năng cơ bản của Firebase miễn phí thông qua gói dịch vụ Spark. Với những tính năng cao hơn, bạn cần phải trả phí khá cao. Bên cạnh đó, cách đăng ký tài khoản Firebase cũng rất đơn giản. Bạn có thể đăng ký tài khoản Firebase thông qua tài khoản Google.
 - + **Dễ sử dụng và tích hợp:** Firebase có giao diện trực quan và thân thiện. Sau khi đăng ký tài khoản, bạn chỉ cần đăng nhập vào là đã có thể sử dụng các tính năng của Firebase. Ngoài ra, Firebase có một kho backend với giao diện đa dạng, phong phú giúp bạn thoải mái lựa chọn.
 - + **Đáp ứng nhu cầu của người dùng:** Firebase chính là một phần của Google. Vì vậy, Google đã không ngừng khai thác và phát triển triệt để những điểm mạnh và các tính năng của Firebase nhằm tạo ra được những dịch vụ đáp ứng nhu cầu của người dùng.
 - + **Cập nhật liên tục và đa nền tảng:** Firebase liên tục cập nhật và có tốc độ phát triển rất nhanh. Nhờ đó mà lập trình viên tiết kiệm được thời gian phát triển và tiếp thị ứng dụng. Bên cạnh đó, Firebase là một nền tảng đa dịch vụ. Điều này được thể hiện thông qua việc Firebase cung cấp đa dạng các dịch vụ với mục đích phát triển website. Qua đó, nhà phát triển có thể dễ dàng lựa chọn được database Firestore hoặc Realtime theo mong muốn.
- **Nhược điểm:**
 - + **Giới hạn về quy mô ứng dụng:** Firebase là một phần Subdomain của Google. Nền tảng này hiện đang bị chặn ở một số quốc gia, trong đó có Trung Quốc. Bên cạnh đó, tất cả cơ sở hạ tầng của nền tảng đều hoạt động trên Google Cloud. Điều

này đồng nghĩa với việc người dùng không thể chạy ứng dụng Firebase trên những đơn vị cung cấp đám mây khác.

- + **Khả năng tùy chỉnh hạn chế:** Firebase chỉ có cơ sở dữ liệu được lưu dưới dạng JSON và không có SQL. Điều này sẽ gây khó khăn trong việc di chuyển và xử lý dữ liệu. Hơn nữa, Firebase không phải là một mã nguồn mở. Do đó, lập trình viên không thể thay đổi được mã nguồn của Firebase, từ đó gây trở ngại cho việc cải tiến sản phẩm.
- + **Bảo mật còn yếu:** Nhược điểm bảo mật còn yếu của Firebase có thể được thể hiện thông qua việc tất cả cơ sở dữ liệu được lưu trữ dưới dạng JSON. JSON là một định dạng nhẹ, không phụ thuộc vào ngôn ngữ và dễ sử dụng. Tuy nhiên, trong một số trường hợp, JSON được đánh giá là độc hại và sẽ xảy ra các vấn đề liên quan đến bảo mật.
- + **Giá các dịch vụ của Firebase:** Hiện nay, Firebase cung cấp cả gói dịch vụ mất phí và miễn phí. Trong đó, Spark đang là gói dịch vụ miễn phí với dung lượng lưu trữ 10GB, SSL cùng với nhiều trang web và các tên miền tùy chỉnh,...

3. Các bước tạo tài khoản:

- **Bước 1:** Truy cập vào trang web chính thức của Google Firebase tại địa chỉ: <https://firebase.google.com/>.
- **Bước 2:** Nhấp vào nút "Bắt đầu" hoặc "Get Started" để bắt đầu quá trình đăng ký.
- **Bước 3:** Đăng nhập bằng tài khoản Google hiện có của bạn hoặc tạo một tài khoản mới nếu bạn chưa có.
- **Bước 4:** Sau khi đăng nhập thành công, bạn sẽ thấy bảng điều khiển chính của Firebase.
- **Bước 5:** Nhấp vào nút "Thêm dự án" hoặc "Add project" để tạo dự án mới.
- **Bước 6:** Đặt tên cho dự án của bạn và chọn quốc gia hoặc khu vực nơi dự án của bạn được đặt.
- **Bước 7:** Tắt chế độ "Google Analytics" nếu bạn không muốn sử dụng tính năng này cho dự án của mình.
- **Bước 8:** Nhấp vào nút "Tạo dự án" hoặc "Create project" để hoàn tất quá trình tạo dự án.
- **Bước 9:** Firebase sẽ tạo dự án mới cho bạn và chuyển bạn đến bảng điều khiển dự án. Tại đây, bạn có thể thêm các dịch vụ Firebase như Realtime Database, Authentication, Cloud Firestore, và nhiều hơn nữa.
- **Bước 10:** Để sử dụng Firebase trong ứng dụng của bạn, bạn cần cung cấp các thông tin cấu hình như khóa API và ID dự án. Các thông tin này có thể được tìm thấy trong phần "Cài đặt dự án" hoặc "Project settings" của bảng điều khiển Firebase.

4. Đọc và giải thích code:

- ✓ Định nghĩa các thuộc tính cơ bản của trang web:

```

<!DOCTYPE html>

<html lang="en"><!--Dinh nghia ngon ngu cua trang web là tieng Anh-->

<head>

    <meta charset="UTF-8"><!--Quy dinh ma ky tu UTF-8-->

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/materialize/1.0.0/css/materialize.min.css">

        <link rel="stylesheet" href="./style.css">

</head>

```

- ✓ Khai báo các thành phần được hiển thị trên giao diện web:

```

<main>

<!-- khung nhiet do va do am -->

<form>

    <h1 id="nhietdo"></h1><!--Khai bao doi tuong tren trang web-->

    <h2 id="doam"></h2><br><!--Khai bao doi tuong tren trang web-->

</form>

<!-- khung hinh anh va cac label -->

<form>

    <!-- hien thi hinh anh -->

```

```

<button type="button" id="btn1" class="btn waves-effect waves-lights">On</button>

<button type="button" id="btn2" class="btn waves-effect waves-lights">Off</button><br><br>

<!-- label hien thi text hoac cac chu dan hien thi tren web -->

<label for="text">This website make by KMT team</label><br>

<label for="text">Nguyen Ngoc Hoang</label><br>

<label for="text">Do Minh Toan</label><br>

<label for="text">Truong Quang Phuc</label><br>

</form>

</main>

```

- ✓ Liên kết với thư viện Firebase JavaScript Core để sử dụng các chức năng cơ bản của Firebase:

```
<script src="https://www.gstatic.com/firebasejs/8.2.10.firebaseio.js"></script>
```

- ✓ Liên kết với thư viện Firebase Realtime Database cho phép bạn làm việc với cơ sở dữ liệu thời gian thực của Firebase:

```
<script src="https://www.gstatic.com/firebasejs/8.2.10.firebaseio.js"></script>
```

- ✓ Dòng này liên kết với thư viện Firebase Analytics cho phép bạn theo dõi và phân tích dữ liệu liên quan đến người dùng và hoạt động trên trang web:

```
<script src="https://www.gstatic.com/firebasejs/8.2.10.firebaseio.analytics.js"></script>
```

- ✓ Cấu hình Firebase:

```
var firebaseConfig = {

  apiKey: "AIzaSyBMo7yeSRaQP5ooUMKs5n4GUb7KIE-bNYU",

  authDomain: "tt-iot-c24a3.firebaseio.com",

  databaseURL: "https://tt-iot-c24a3-default-rtdb.firebaseio.com",

  projectId: "tt-iot-c24a3",
```

```
storageBucket: "tt-iot-c24a3.appspot.com",
messagingSenderId: "442039349477",
appId: "1:442039349477:web:b3b9f016980b6e367ce40c",
measurementId: "G-37MY2WT5JV"
};
```

- ✓ Khởi tạo Firebase và bật tính năng Firebase Analytics trong ứng dụng web:

```
firebase.initializeApp(firebaseConfig);
firebase.analytics();
```

- ✓ Cập nhật dữ liệu từ Firebase về trang web:

```
var nhietDo = document.getElementById('nhietdo');

var dbRef = firebase.database().ref().child('Nhiệt do');

var doAm = document.getElementById('doam');

var dbRef2 = firebase.database().ref().child('Đo am');

dbRef.on('value', snap => nhietDo.innerText = snap.val());

dbRef2.on('value', snap => doAm.innerText = snap.val());
```


CHƯƠNG 3: THIẾT KẾ HỆ THỐNG

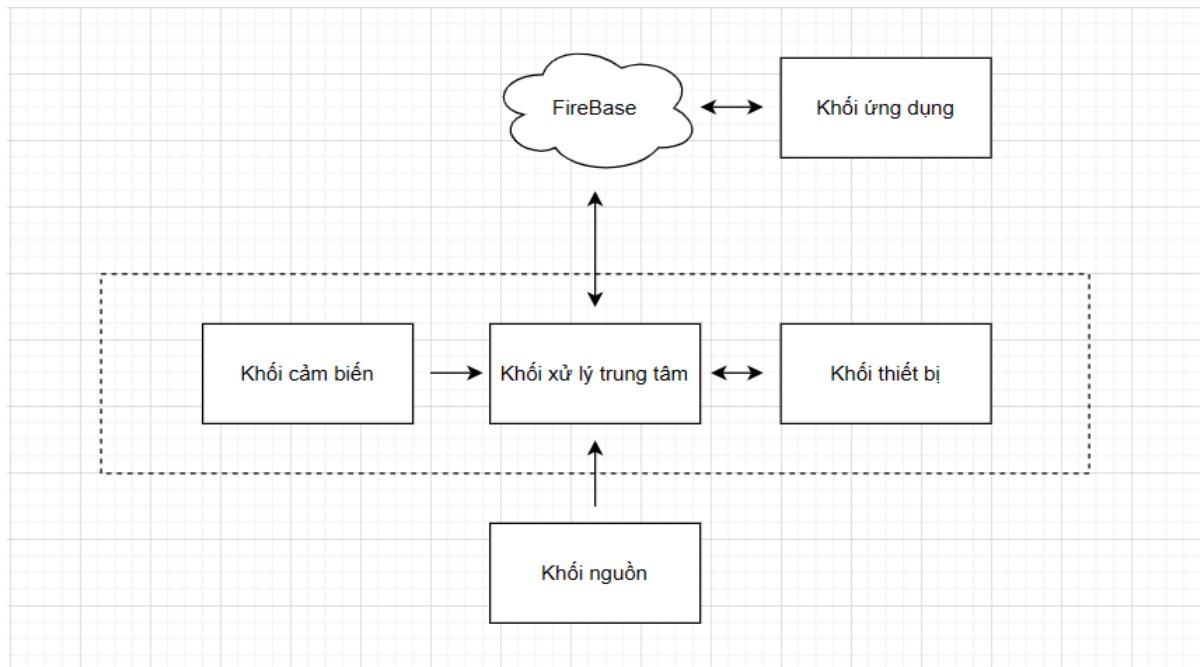
1. Yêu cầu thiết kế:

Hệ thống Smart Farm được thiết kế để phục vụ việc quản lý các thông số môi trường của khu vườn một cách hiệu quả và chính xác nhất. Để đáp ứng được các tiêu chí đó, hệ thống cần đáp ứng được các yêu cầu sau:

- Hệ thống cần được thiết kế một cách hợp lý để các cảm biến như nhiệt độ, ánh sáng, độ ẩm,... có thể đọc được các dữ liệu từ môi trường một cách chính xác nhất.
- Hệ thống cần đảm bảo được khả năng lưu trữ và truyền những dữ liệu thu thập được từ các điểm tới cơ sở dữ liệu. Đồng thời các dữ liệu từ cơ sở dữ liệu cần được truyền đến các điểm để thực hiện các yêu cầu của hệ thống.
- Hệ thống cần hiển thị được đầy đủ các thông tin cho người dùng một cách trực quan và chính xác nhất, các tính năng điều khiển cần được thiết kế đơn giản, dễ sử dụng nhưng vẫn phải đảm bảo các chức năng.
- Trực quan hóa các dữ liệu thông qua biểu đồ để dễ dàng theo dõi các sự thay đổi của các thông số môi trường.

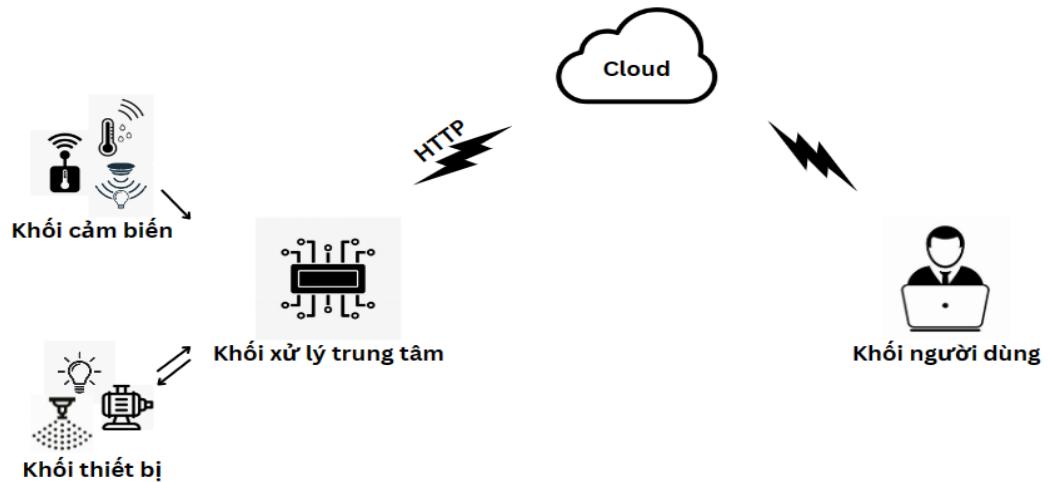
2. Sơ đồ hệ thống

a. Sơ đồ khối:



Hình 1: Sơ đồ khối

b. Sơ đồ đặc tả hệ thống:



Hình 2: Sơ đồ đặc tả hệ thống

Khối cảm biến: Bao gồm các cảm biến nhiệt độ, ánh sáng, độ ẩm thực hiện chức năng cập nhật các thông số từ môi trường truyền về cho khối xử lý trung tâm.

Khối xử lý trung tâm: Là vi điều khiển ESP32 thực hiện chức năng nhận các thông tin từ cảm biến, xử lý và cập nhật lên database, cập nhật và điều khiển trạng thái của khối thiết bị thông qua yêu cầu của người dùng.

Khối thiết bị: Bao gồm các thiết bị đèn, máy bơm và máy phun sương được điều khiển để can thiệp hoặc thay đổi các thông số của môi trường thông qua sự điều khiển của khối xử lý trung tâm.

Khối Cloud: Cơ sở dữ liệu (ở đây là Google FireBase) được sử dụng để lưu trữ các dữ liệu thu thập được thông qua đó sử dụng nó để phục vụ các chức năng của hệ thống.

Khối người dùng: Là những người sử dụng các chức năng của hệ thống, có thể trực tiếp điều khiển các chức năng mà hệ thống cung cấp để phục vụ cho yêu cầu sử dụng.

Giao thức truyền tin: Ở đây giao thức chính được sử dụng là HTTP. Giao thức HTTP hoạt động dựa trên mô hình Client – Server. Trong mô hình này, các máy tính của người dùng sẽ đóng vai trò làm máy khách (Client). Sau một thao tác nào đó của người dùng, các máy khách sẽ gửi yêu cầu đến máy chủ (Server) và chờ đợi câu trả lời từ những máy chủ này. [1]

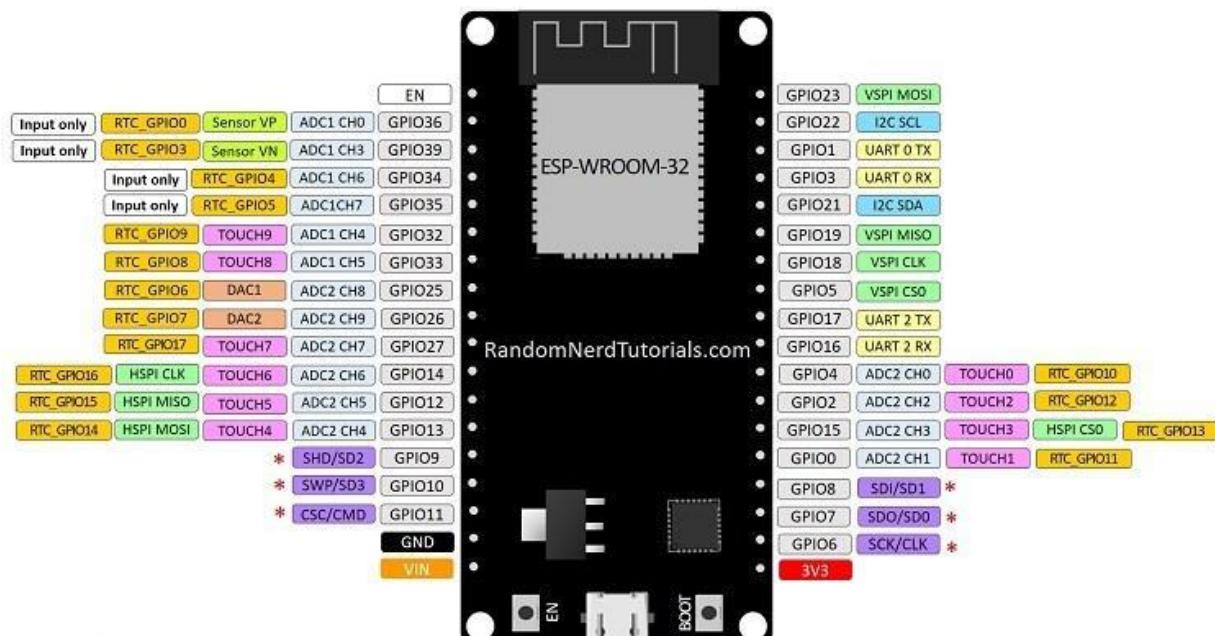
c. Giới thiệu phần cứng:

- Khối xử lý trung tâm:

Vi điều khiển ESP32 WROOM là một vi điều khiển tích hợp Wi-Fi và Bluetooth, được phát triển bởi công ty Espressif Systems và được sử dụng rộng rãi trong các ứng dụng IoT và các hệ thống nhúng. [2]

ESP32 DEVKIT V1 – DOIT

version with 36 GPIOs



Hình 3: Sơ đồ chân ESP32

Các điểm chính của vi điều khiển ESP32 WROOM bao gồm:

- + *Tích hợp Wi-Fi và Bluetooth: ESP32 WROOM có khả năng kết nối với mạng Wi-Fi và các thiết bị Bluetooth, giúp cho việc truyền dữ liệu không dây trở nên dễ dàng hơn.*
- + *Xử lý mạnh mẽ: ESP32 WROOM có khả năng xử lý mạnh mẽ, với hai nhân xử lý tốc độ cao và bộ nhớ lớn, giúp cho việc xử lý dữ liệu trở nên nhanh chóng và hiệu quả hơn.*
- + *Hỗ trợ nhiều loại cảm biến: ESP32 WROOM có khả năng kết nối với nhiều loại cảm biến khác nhau, giúp cho việc thu thập dữ liệu từ môi trường xung quanh trở nên dễ dàng hơn.*
- + *Hỗ trợ nhiều giao thức truyền thông: ESP32 WROOM hỗ trợ nhiều giao thức truyền thông khác nhau, bao gồm SPI, I2C, UART và CAN, giúp cho việc kết nối với các thiết bị khác trở nên dễ dàng hơn.*

- + *Tính năng tiết kiệm năng lượng: ESP32 WROOM có khả năng tiết kiệm năng lượng, giúp cho việc sử dụng pin kéo dài hơn và làm cho nó trở thành một lựa chọn tốt cho các ứng dụng yêu cầu tiết kiệm năng lượng.*
- ***Khối cảm biến:***
 - ***Cảm biến nhiệt độ và độ ẩm DHT11:***

DHT11 là một cảm biến nhiệt độ và độ ẩm đơn giản và phổ biến được sử dụng trong các ứng dụng IoT, mạng cảm biến và các hệ thống điều khiển tự động. DHT11 có các thông số kỹ thuật như sau:

- + Độ chính xác đo nhiệt độ: $\pm 2^{\circ}\text{C}$.
- + Độ chính xác đo độ ẩm: $\pm 5\%$.
- + Khoảng đo nhiệt độ: 0 đến 50°C
- + Khoảng đo độ ẩm: 20% đến 90%.
- + Điện áp hoạt động: 3V đến 5.5V.
- + Dòng hoạt động tối đa: 2.5mA. [3]

- ***Module cảm biến ánh sáng LDR:***

Module cảm biến ánh sáng là một linh kiện điện tử có điện trở thay đổi giảm theo ánh sáng chiếu vào. Quang trở làm bằng chất bán dẫn trở kháng cao, và không có tiếp giáp nào. Trong bóng tối, quang trở có điện trở đến vài $M\Omega$. Khi có ánh sáng, điện trở giảm xuống mức một vài trăm Ω . Một số thông số kỹ thuật như sau:

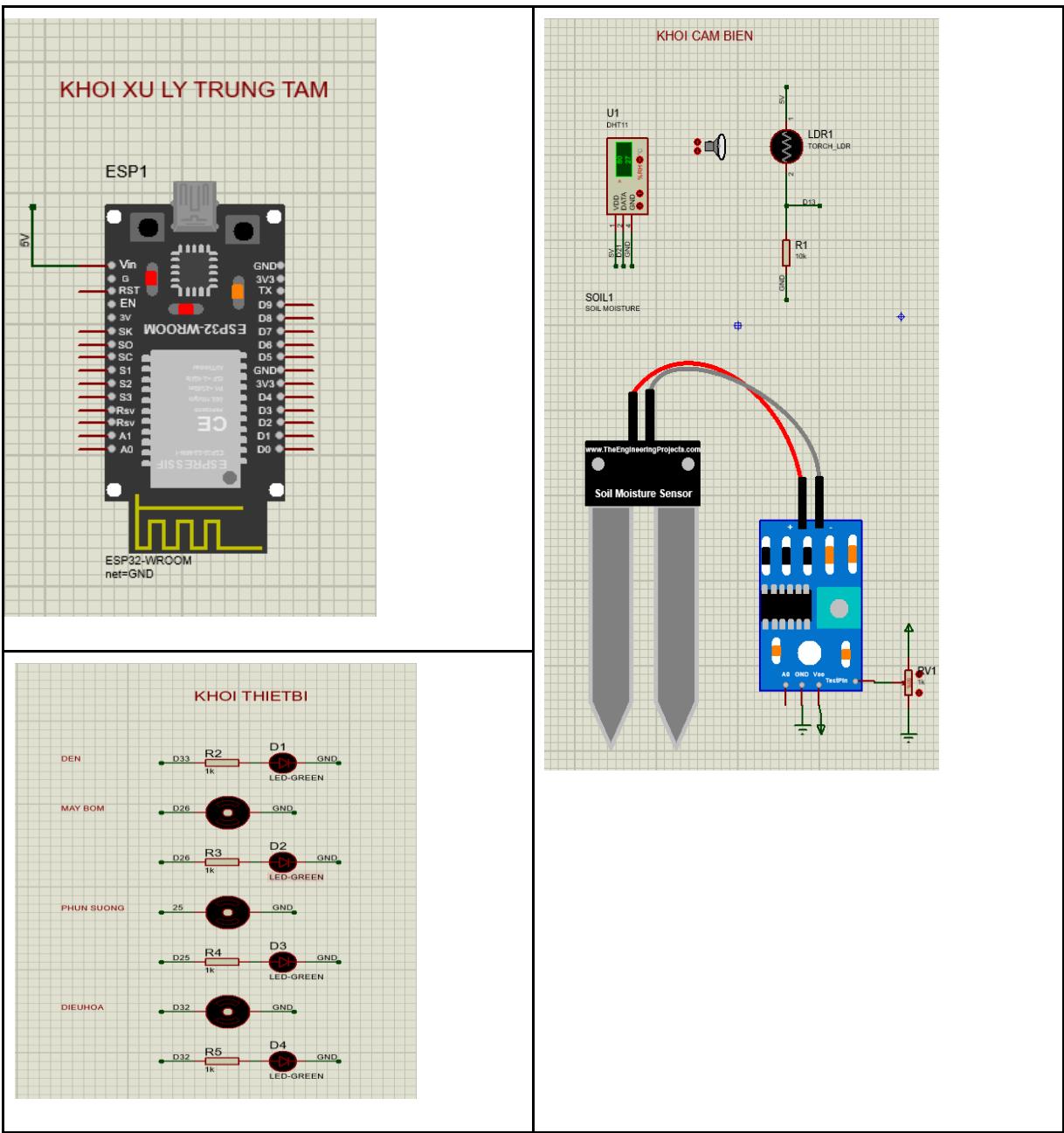
- + Điện áp hoạt động: 3.3V-5V.
- + IC so sánh : LM393.
- + DO: Đầu ra tín hiệu số (0 và 1).
- + Kích thước PCB: 3cm * 1.6cm.

- ***Cảm biến độ ẩm đất Moisture Soil V2.0:***

Cảm biến độ ẩm đất là một giải pháp được ứng dụng trong lĩnh vực nông nghiệp như chăm sóc cây trồng, giúp giám sát và điều chỉnh mức độ ẩm trong đất một cách chính xác và hiệu quả. Việc sử dụng cảm biến đất không chỉ mang lại lợi ích về tiết kiệm nước và tối ưu hóa việc tưới tiêu, mà còn giúp nâng cao hiệu suất cây trồng và bảo vệ môi trường. Một số thông số kỹ thuật như sau:

- + Điện áp hoạt động: 3.3V-5V.
- + Dòng hoạt động tối đa: 5mA.
- + Điện áp ra: 1.2V-3V.

- d. ***Sơ đồ kết nối:***



Hình 4: Sơ đồ kết nối

- Khối xử lý trung tâm:

- + ESP32 là vi điều khiển chính của hệ thống. Nó thu thập dữ liệu từ các cảm biến, quản lý các thiết bị (LED và buzzer), và gửi dữ liệu này lên Firebase Cloud.

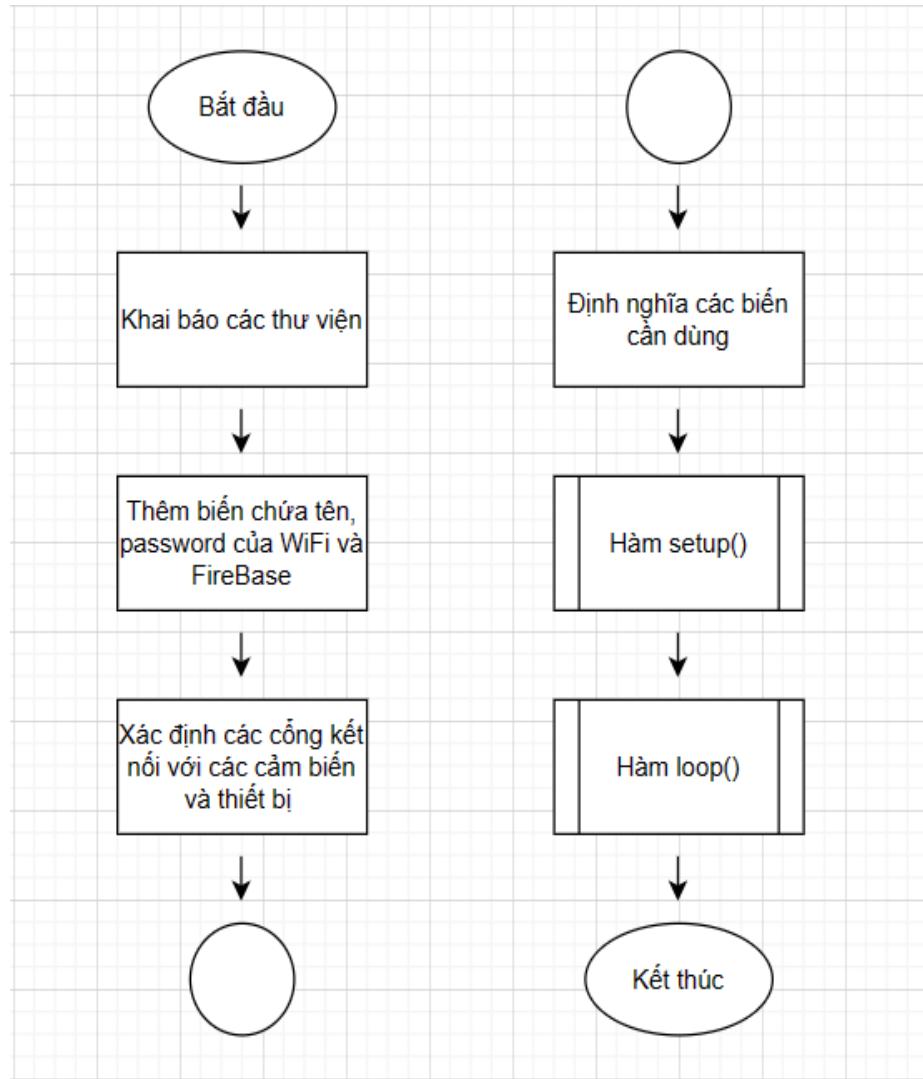
- Khối cảm biến:

- + Cảm biến DHT11: Dùng để đo nhiệt độ và độ ẩm trong môi trường. Thông tin này có thể được sử dụng để kiểm tra điều kiện môi trường. DHT11 được kết nối với Khối xử lý qua chân D21.
- + Cảm biến ánh sáng LDR: Dùng để cảm nhận ánh sáng của môi trường. Thông tin này dùng để đánh giá điều kiện của môi trường ảnh hưởng đến sự phát triển của cây trồng. Cảm biến LDR được kết nối với Khối xử lý thông qua chân D13.

- + Cảm biến độ ẩm đất Soil Moisture: Dùng để đo độ ẩm của đất trồng. Dùng để đánh giá điều kiện sống của cây. Cảm biến độ ẩm đất được kết nối với Khối xử lý thông qua chân D35.
- ***Khối thiết bị:***
 - + DEN: đèn dùng để kiểm soát giá trị ánh sáng của môi trường. DEN được điều khiển qua chân D33 của Khối xử lý.
 - + MAYBOM: máy bơm nước dùng để kiểm soát giá trị độ ẩm của đất trồng, khi giá trị đó xuống đến thấp hơn so với yêu cầu của cây trồng thì nó sẽ được điều khiển để thay giá trị đó. MAYBOM được Khối xử lý điều khiển thông qua chân D26.
 - + PHUNSUONG: hệ thống van phun sương để kiểm soát giá trị độ ẩm không khí, khi giá trị đó xuống đến thấp hơn so với yêu cầu của cây trồng thì nó sẽ được điều khiển để thay giá trị đó. PHUNSUONG được Khối xử lý điều khiển thông qua chân D25.
 - + DIEUHOA: điều hoà dùng để kiểm soát nhiệt độ của SmartFarm. DIEUHOA được kết nối thông qua chan D32.

3. Thiết kế phần mềm:

a. Lưu đồ hệ thống



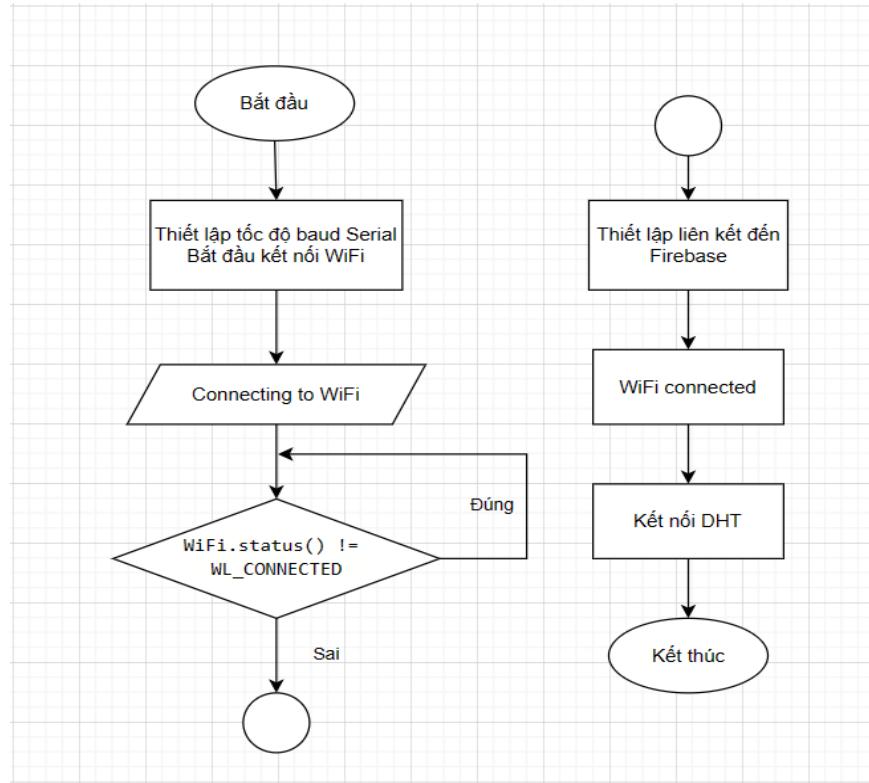
Hình 5: Lưu đồ hệ thống

- Bắt đầu: bắt đầu chương trình
- Khai báo thư viện: khai báo các thư viện cần dùng như:


```
#include <WiFi.h>
#include <HTTPClient.h>
#include <DHT.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
```
- Thêm biến chứa tên, password của WiFi và Firebase:
 - + Tạo các biến chứa địa chỉ và password của WiFi
 - + Tạo biến chứa địa chỉ url của Firebase và API key của Firebase cần liên kết
- Xác định các cổng kết nối cho các cảm biến và thiết bị

- Tạo các biến cần dùng để chứa giá trị đo được từ cảm biến, trạng thái led hoặc các biến dùng để lưu lại và so sánh trong bài
- Hàm setup(): chi tiết bên dưới
- Hàm loop(): chi tiết bên dưới

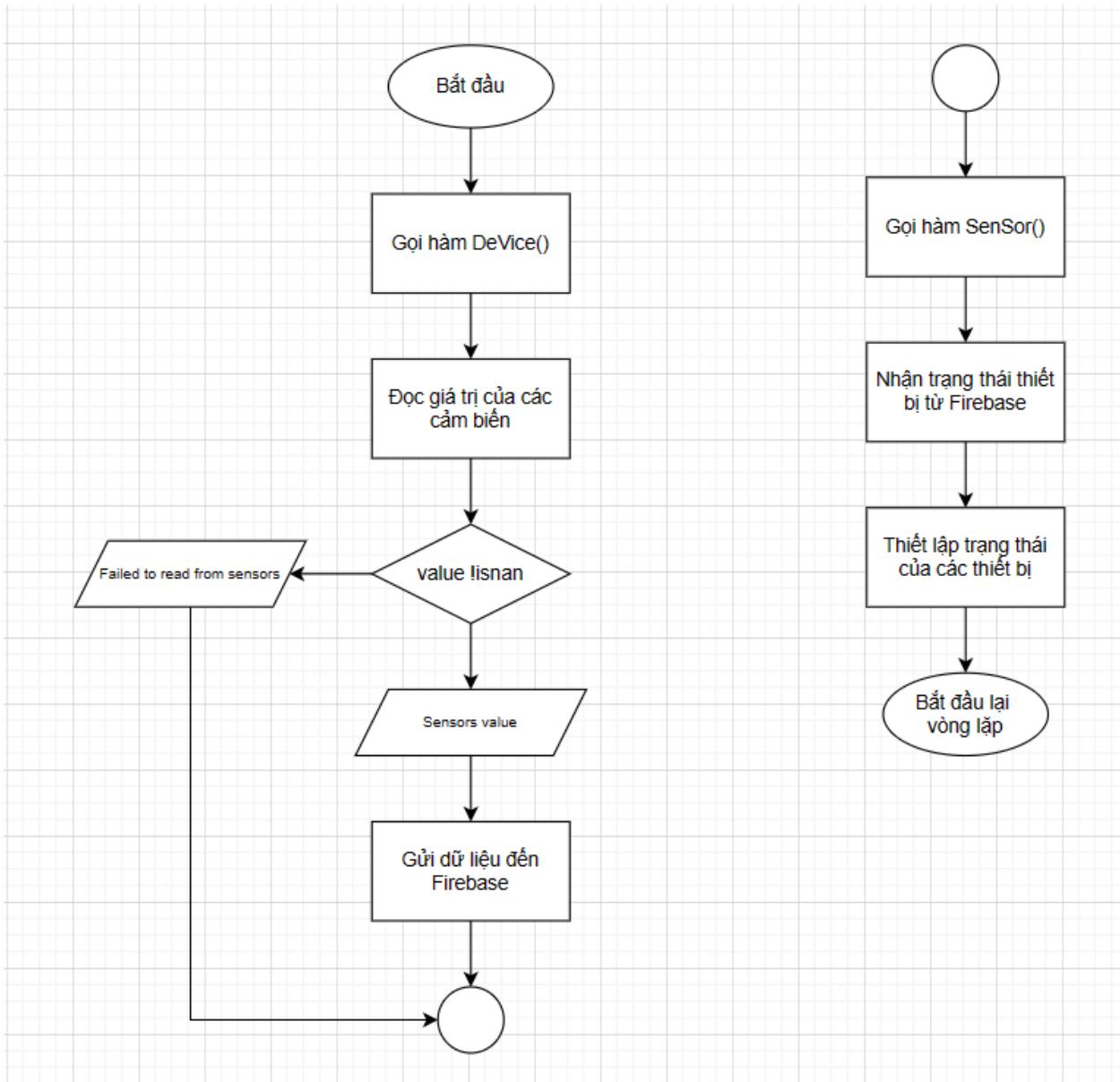
b. *Hàm setup()*



Hình 6: Lưu đồ hàm setup

- Bắt đầu: bắt đầu hàm setup()
- Thiết lập tốc độ Baud của Serial: chọn tốc độ baud 115200
- Kết nối wifi: kết nối wifi dùng lệnh Wifi.begin(biến địa chỉ, biến password)
- Wifi.status()!=WL_CONNECTED: kiểm tra xem wifi có không kết nối ESP32 không
 - + Nếu kiểm tra đúng (wifi chưa được kết nối): lặp lại đèn khi kết nối
 - + Nếu kiểm tra sai (wifi đã kết nối): tiếp tục
- In qua serial Wifi connected
- Thiết lập liên kết tới Firebase
- Kết nối DHT: gọi hàm dht.begin() để bắt đầu kết nối với cảm biến DHT
- Kết thúc: kết thúc hàm setup()

c. *Hàm loop()*

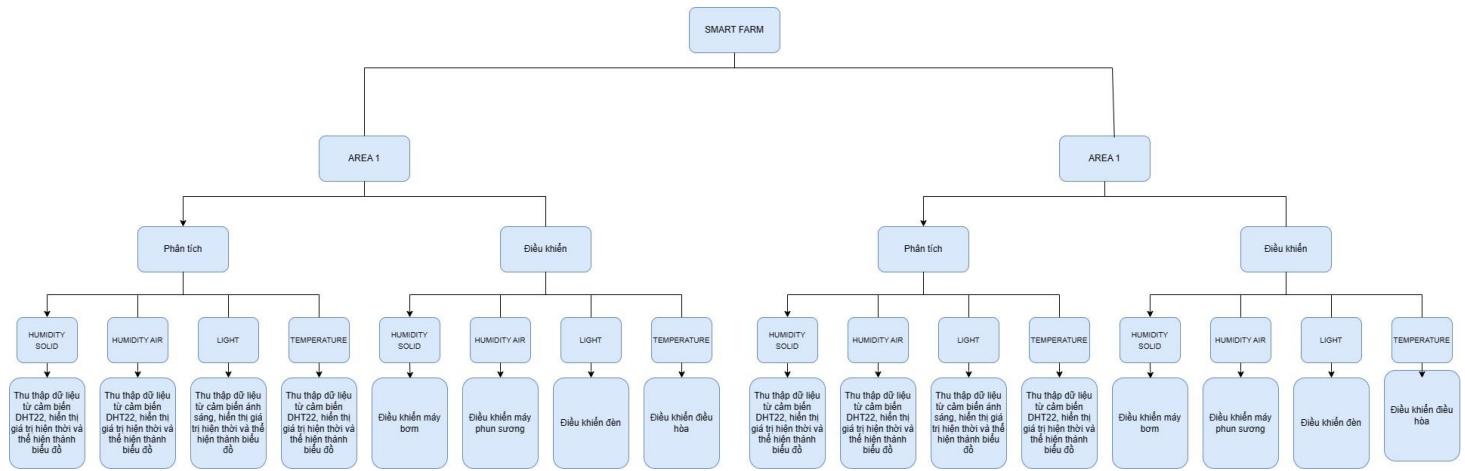


Hình 7: Lưu đồ hàm loop

- Bắt đầu: Bắt đầu hàm loop().
- Gọi hàm DeVice(): để chương trình thực hiện các lệnh trong hàm.
- Đọc giá trị của các cảm biến: thực hiện nhận các giá trị từ cảm biến của hệ thống.
- Kiểm tra xem giá trị có được đọc đúng hay không:
 - + Nếu có: thực hiện câu lệnh tiếp theo.
 - + Nếu không: thông báo không thành công và kết thúc hàm.
- Gọi hàm SenSor(): để chương trình thực hiện các lệnh trong hàm.
- Nhận trạng thái thiết bị từ Firebase: thực hiện nhận các dữ liệu về trạng thái của các thiết bị.
- Thiết lập trạng thái của các thiết bị: từ các dữ liệu nhận được thực hiện điều khiển các thiết bị.
- Bắt đầu lại vòng lặp: Bắt đầu lại hàm loop().

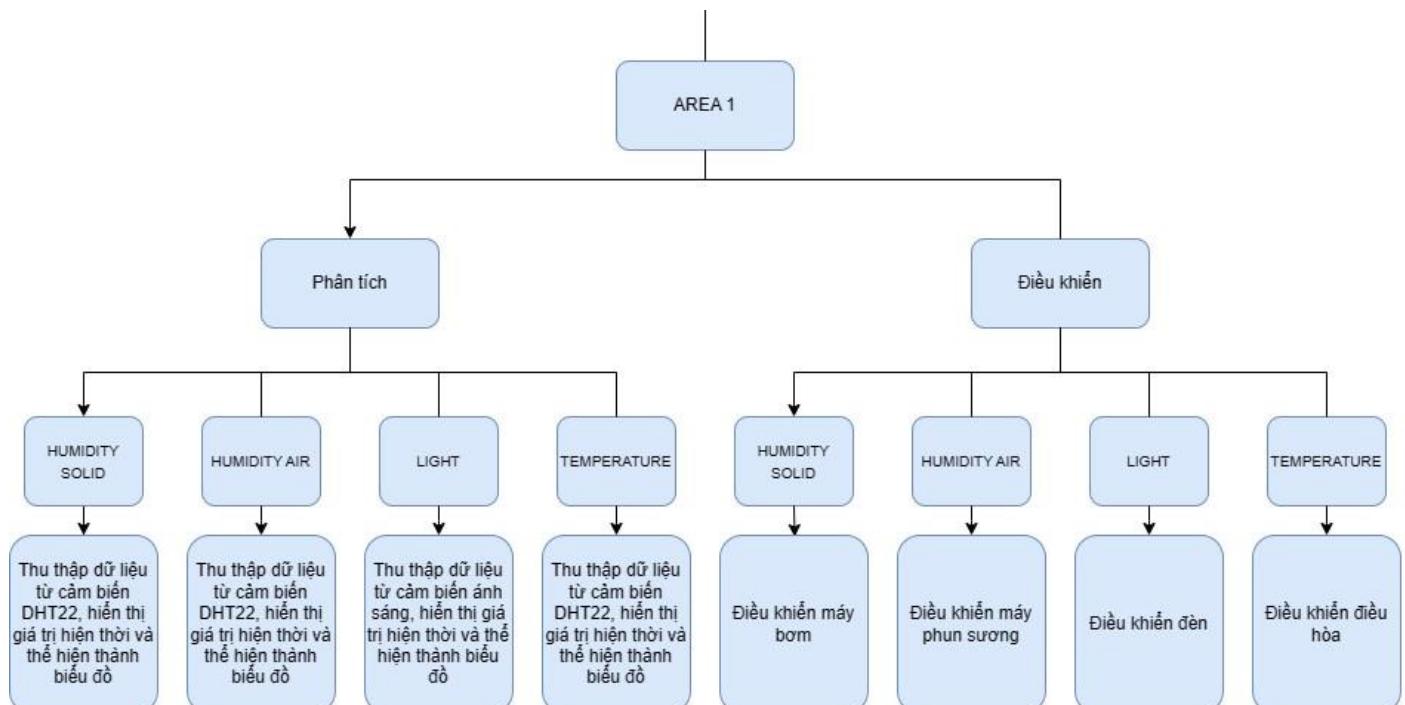
d. Lưu đồ web

Lưu đồ tổng thể



Hình 8: Lưu đồ web tổng thể

Lưu đồ trong từng khu vực



Hình 9: Lưu đồ web từng khu vực

4. Kết quả

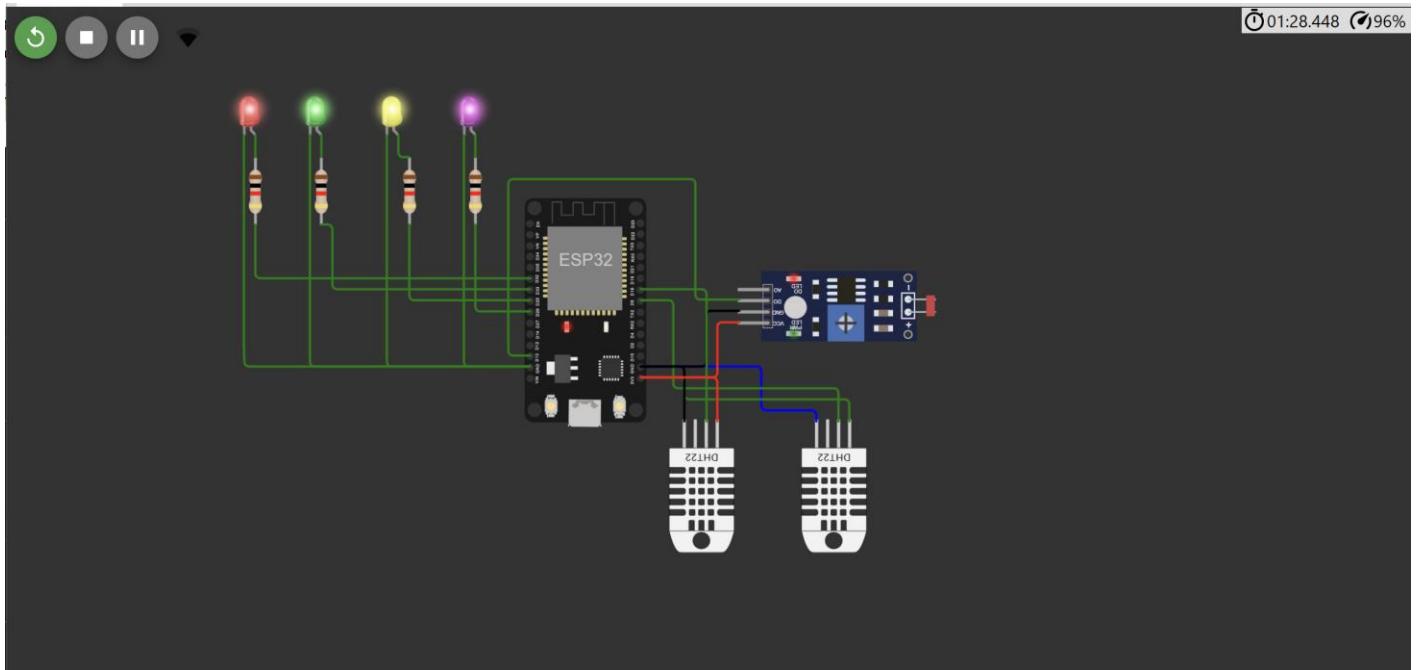
a. Giao diện Web:

Đây là hình ảnh web khi hoàn thiện, khi load web sẽ có một số hiệu ứng. Giao diện đáp ứng đủ các tiêu chí dễ nhìn, thể hiện đầy đủ thông tin và có thể điều khiển phần cứng.



Hình 10: Hình ảnh web hoàn thiện

b. Mô phỏng:

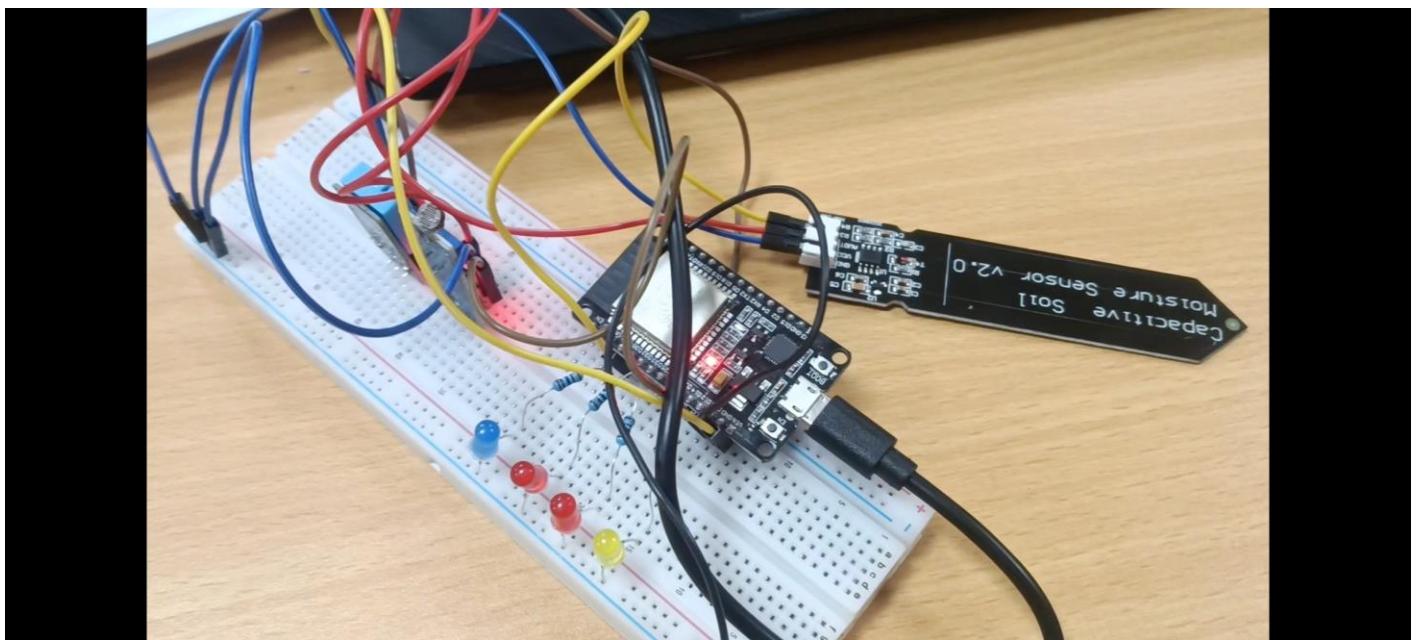


Hình 11: Mô phỏng phần cứng

Đây là mô phỏng được thực hiện trên trang web <https://wokwi.com/> sử dụng esp32 và các phần cứng liên quan. Và sử dụng giao thức HTTP để truyền và nhận dữ liệu từ Firebase

c. Phần cứng:

Lập trình phần cứng sử dụng phần mềm Arduino IDE và các thư viện do nó cung cấp để điều khiển, liên kết với Firebase.



Hình 12: Phần cứng

Link Web: https://khalin0401.github.io/TT_IoT/

Link mô phỏng: <https://wokwi.com/projects/382489033222942721>

Link video Demo: <https://youtu.be/kbtIBlmMFtM>

5. Kết luận:

a. Trong một học kỳ vừa nghiên cứu và thực hiện đề tài thì nay chúng em đã hoàn thiện được đề tài với những kết quả đạt được sau đây:

- ✓ Hệ thống thu thập được dữ liệu từ các cảm biến và đưa dữ liệu về Google Firebase với độ trễ tương đối thấp.
- ✓ Xây dựng được giao diện website, có thể hiển thị các thông số đo được và điều khiển các thiết bị.
- ✓ Hệ thống hoạt động ổn định, đồng bộ giữa web và phần cứng, thực hiện được các chức năng và thỏa mãn các yêu cầu đề ban đầu.

b. Hướng phát triển tương lai:

- ✓ Chúng em đưa ra một số đề xuất nhằm cải tiến để tăng tính ứng dụng và hoàn thiện hệ thống hơn:
- ✓ Tăng số lượng cảm biến, thiết bị điều khiển để phục vụ tốt hơn cho nhu cầu của SmartFarm.
- ✓ Xây dựng thêm hệ thống quản lý chất lượng nước để cung cấp cho cây trồng.
- ✓ Thay đổi giao thức truyền nhận dữ liệu giữa các cảm biến và khói xử lý để đạt được hiệu suất tốt nhất như các giao thức MQTT, Lora,vv.
- ✓ Nguyên cứu để tích hợp các mô hình máy học để phân tích dữ liệu nhận được từ cảm biến đồng thời đưa ra các phương án gợi ý để cây trồng đạt năng suất cao nhất.

TÀI LIỆU THAM KHẢO

- [1] Đ. Đạt, “Tìm hiểu về HTTP (HyperText Transfer Protocol),” [Trực tuyến]. Available: <https://viblo.asia/p/tim-hieu-ve-http-hypertext-transfer-protocol-bJzKmgewl9N>.
- [2] wikipedia, “wikipedia,” [Trực tuyến]. Available: <https://vi.wikipedia.org/wiki/ESP32>.
- [3] Vngiotlab, “Đo nhiệt độ, độ ẩm từ cảm biến DHT11,” [Trực tuyến]. Available: https://vngiotlab.github.io/vbluno/vi/mydoc_arduino_tut10_vi.html.

Ảnh thành viên nhóm

