

## Project Documentation

|              |   |
|--------------|---|
| Date         | 3 November 2023                         |
| Team ID      | NM2023TMID02678                         |
| Project Name | How to create a landing page in Hubspot |

### Ideation Phase:

#### Define the Problem Statement:

#### How To Create A Landing page In Hubspot:

Many businesses do not have the in-house expertise to create effective landing pages using HubSpot. This can lead to landing pages that are poorly designed, cluttered, and ineffective at converting visitors.

Creating a landing page can be a time-consuming process, especially if you are not familiar with HubSpot. This can be a challenge for businesses that are short on resources.

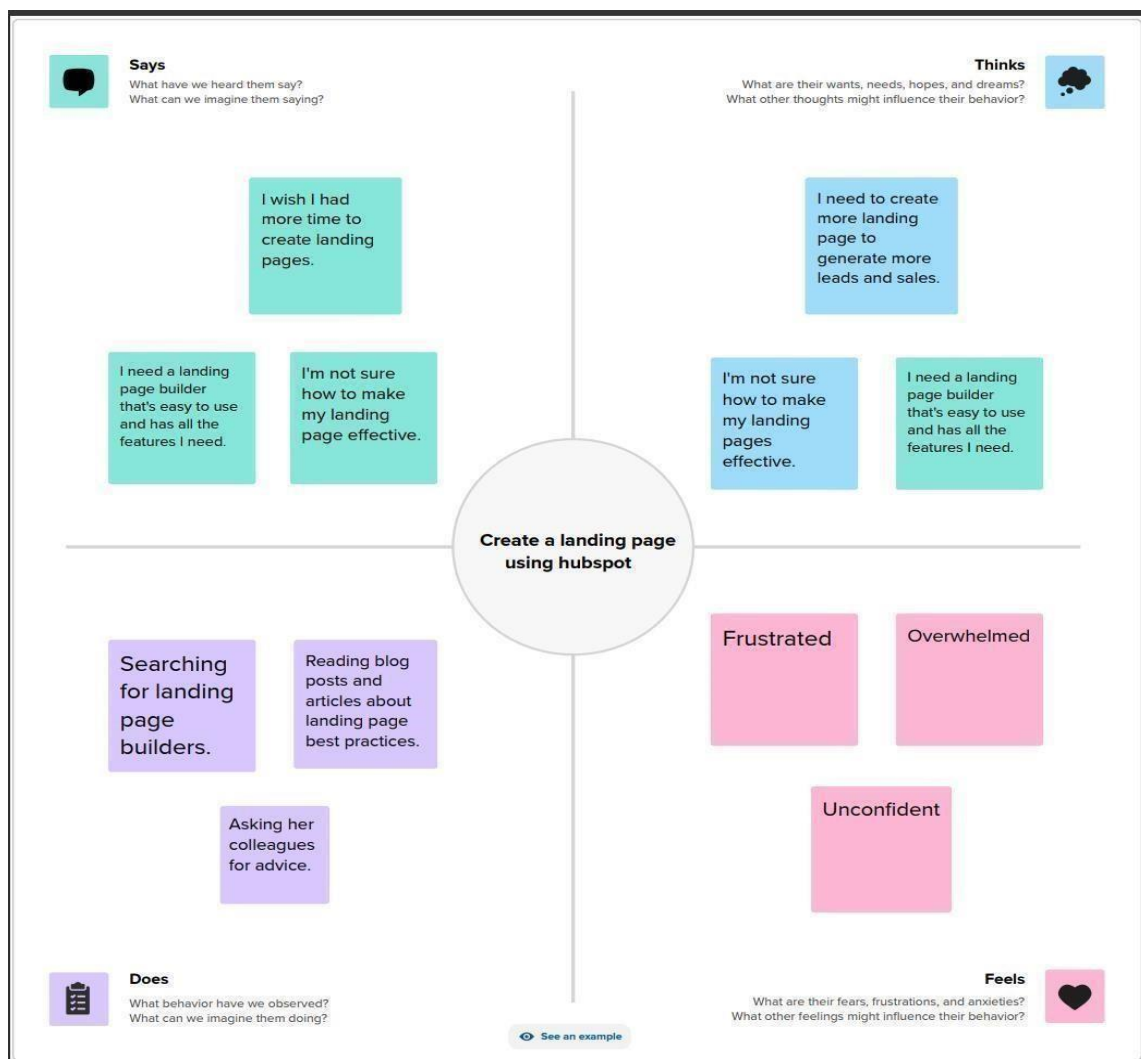
HubSpot's landing page templates can be limiting for businesses that want to create highly customized landing pages. This can make it difficult to create landing pages that are unique and stand out from the competition.

| <b>Problem Statement(PS)</b> | <b>I am (Customer) to</b> | <b>I'm trying</b>                | <b>But</b>                       | <b>Because</b>                  | <b>Which makes me feel</b> |
|------------------------------|---------------------------|----------------------------------|----------------------------------|---------------------------------|----------------------------|
| Performance issues           | Customer                  | Sign up for the services         | It does not support              | It takes long time to load      | Frustrated                 |
| Subscription cost            | Student                   | Finding templates for my project | There are limited free templates | The premium versions are costly | Disappointed               |

|                 |              |                        |                             |                         |                         |
|-----------------|--------------|------------------------|-----------------------------|-------------------------|-------------------------|
| Version control | Professional | Work on my new project | It needs collaborative work | It has limited versions | Difficult to contribute |
|-----------------|--------------|------------------------|-----------------------------|-------------------------|-------------------------|

## Empathy Map:


Use this framework to develop a deep, shared understanding and empathy for other people. An empathy map helps describe the aspects of a user's experience, needs and pain points, to quickly understand your users' experience and mindset.



## Brainstrom & Idea Prioritization:

Brainstorming provides a free and open environment that encourages everyone within a team to participate in the creative thinking process that leads to problem solving. Prioritizing volume over value, out-of-the-box ideas are welcome and built upon, and all participants are encouraged to collaborate, helping each other develop a rich amount of creative solutions.

Template



### Brainstorm & idea prioritization

Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.

⌚ 10 minutes to prepare  
👥 1 hour to collaborate  
👤 2-8 people recommended

➔

#### Before you collaborate

A little bit of preparation goes a long way with this session. Here's what you need to do to get going.

⌚ 10 minutes

1

Team gathering

Define who should participate in the session and send an invite. Share relevant information or pre-work ahead.

2

Set the goal

Think about the problem you'll be focusing on solving in the brainstorming session.

3

Learn how to use the facilitation tools

Use the Facilitation Superpowers to run a fun and productive session.

[Open article](#) ➔

#### Define your problem statement

Creating a landing page in HubSpot can be challenging, especially for users who are new to the platform. There are a number of different factors to consider, such as designing the page, writing the copy, and setting up the lead capture form. Additionally, it can be difficult to track the performance of landing pages in HubSpot, which can make it difficult to identify which elements are working and which ones need to be improved.

⌚ 5 minutes

PROBLEM

How might we [your problem statement]?

Key rules of brainstorming

To run a smooth and productive session:

🗨️ Stay in topic.

💡 Encourage wild ideas.

🙊 Defer judgment.

👂 Listen to others.

🗨️ Go for volume.

👁️ If possible, be visual.

2

## Brainstorm

Write down any ideas that come to mind that address your problem statement.

🕒 10 minutes

### TIP

You can select a sticky note and hit the pencil switch to sketch! (or to start drawing!)

### Kavitha.M

Determine the primary purpose of your landing page.

Identify your target audience.

Understand the needs and preferences of audience.

Craft a clear and attention grabbing headline.

Use relevant images and videos.

Design a user friendly form.

### Keerthiga.L.K

User friendly form should be designed.

Keep the landing page focused.

Plan the post submission experience with a thank you page.

Setup hubspot tracking.

Decide how you will drive traffic.

Monitor anding page performance.

### Ishwariya.A

Target audience should be identified.

Create a prominent call to action button.

Include customer testimonials.

Attention grabbing headline should be included.

Ensure your landing pages works well on mobile.

Incorporate relevant keywords.

### Siva Ranjini.S.L

User friendly form is needed.

Target audience should be identified.

Keep your landing page without distraction.

Images and videos are included.

Determine the purpose of your landing page.

Use relevant keywords.

3

## Group ideas

Take turns sharing your ideas while clustering similar or related notes as you go. Once all sticky notes have been grouped, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you can break it up into smaller sub-groups.

🕒 20 minutes

### TIP

And encourage everyone to sticky notes to make it easier to find, browse, organize, and categorize important ideas as themes within your mind.

Design a user friendly form.

Use relevant images and videos.

Incorporate relevant keywords.

Keep your landing page focused.

User friendly form is needed.

Images and videos are included.

Use relevant keywords.

Keep your landing page without distraction.

User friendly form should be designed.



|  |  |  |
|--|--|--|
|  |  | difficult to identify which elements are working and which ones need to be improved. |
|--|--|--|

|    |                           |   |
|----|---------------------------|---|
| 2. | Idea/Solution description | The landing page builder could be implemented as a standalone application or as a HubSpot plugin. The standalone application would be available to all HubSpot users, regardless of their subscription plan. The HubSpot plugin would be available to users of the Marketing Hub Professional and Enterprise plans.   |
| 3. | Novelty/Uniqueness        | The landing page builder would be the first landing page builder to be integrated with HubSpot analytics. This integration would allow users to track the performance of their landing pages directly from the landing page builder, without having to switch to a different tool. The landing page builder would provide users with a library of pre-written landing page copy. This library would be the first of its kind to be integrated with a landing page builder. The landing page builder would be focused on helping users to create landing pages that convert visitors into leads and customers. |

|    |                                      |   |
|----|--------------------------------------|---|
| 4. | Social Impact/Customer Satisfication | <p>A small business owner could use the landing page builder to create a landing page for a new product or service. The landing page could help the business owner to generate more leads and sales for the new product or service. A startup could use the landing page builder to create a landing page for its crowdfunding campaign. The landing page could help the startup to raise the money it needs to launch its business. A non-profit organization could use the landing page builder to create a landing page for a donation campaign. The landing page could help the non-profit organization to raise more money for its cause.</p>  |
| 5. | Business Model(Revenue Model)        | <p>The landing page builder would be offered in three different subscription tiers: Basic, Professional, and Enterprise. The Basic tier would provide users with access to the core features of the landing page builder, such as the drag-and-drop editor, the library of predesigned landing page templates, and the lead capture form builder. The Professional tier would provide users with access to all of the features of the Basic tier, plus additional features such as the library of pre-written landing page copy and the integration with HubSpot analytics. The Enterprise tier would provide users with access to all of the features of the Professional tier, plus additional features such as custom branding and priority support.</p> |

|    |                           |  |
|----|---------------------------|--|
| 6. | Scability of the Solution | A small business with a handful of landing pages could start with the Basic tier of the landing page builder. As the business grows and needs to create more complex landing pages, it could upgrade to the Professional or Enterprise tier. A large enterprise with hundreds or thousands of landing pages could use the Enterprise tier of the landing page builder to create and manage its landing pages at scale. A marketing agency that creates landing pages for its clients could use the landing page builder to create and manage landing pages for all of its clients. |
|----|---------------------------|--|

### **Solution Architecture:**

Creating a solution architecture block diagram involves visually representing the components and interactions of a system or solution.

At the top of your diagram, include a title and a brief description to explain the purpose of the architecture.

Show any external interfaces that the solution interacts with, such as users, thirdparty services, or external systems. These are typically depicted as labeled arrows entering and exiting the diagram.

Here are some specific examples of how the landing page builder could have a positive social impact and customer satisfaction:

- A small business owner could use the landing page builder to create a landing page for a new product or service. The landing page could help the business owner to generate more leads and sales for the new product or service.
- A startup could use the landing page builder to create a landing page for its crowdfunding campaign. The landing page could help the startup to raise the money it needs to launch its business.
- A non-profit organization could use the landing page builder to create a landing page for a donation campaign. The landing page could help the non-profit organization to raise more money for its cause.



In all of these cases, the landing page builder would be helping businesses and organizations to achieve their goals. This would have a positive impact on the economy and society as a whole.

### Example–Solution Architecture Diagram:

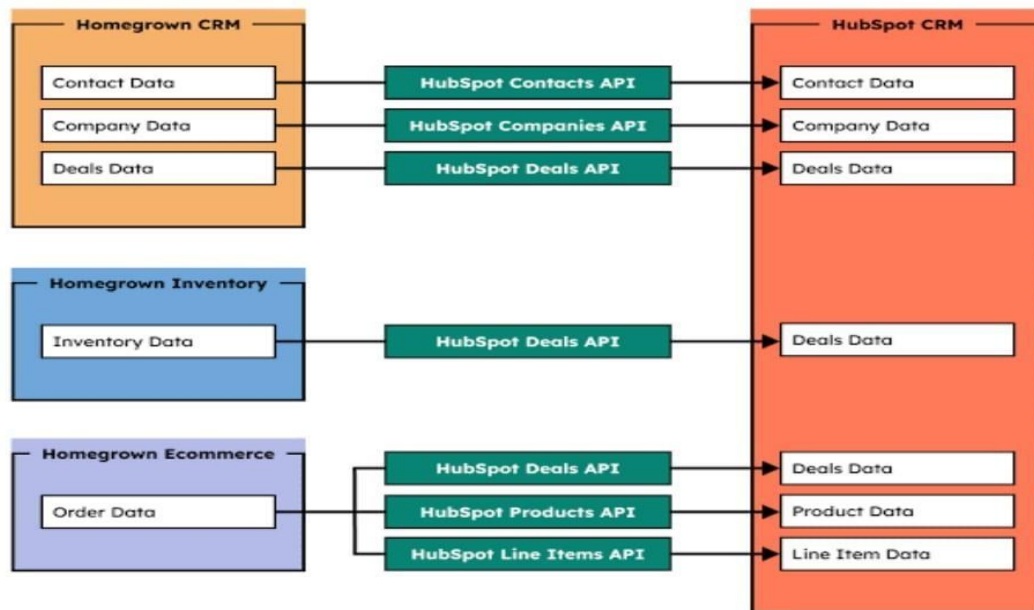
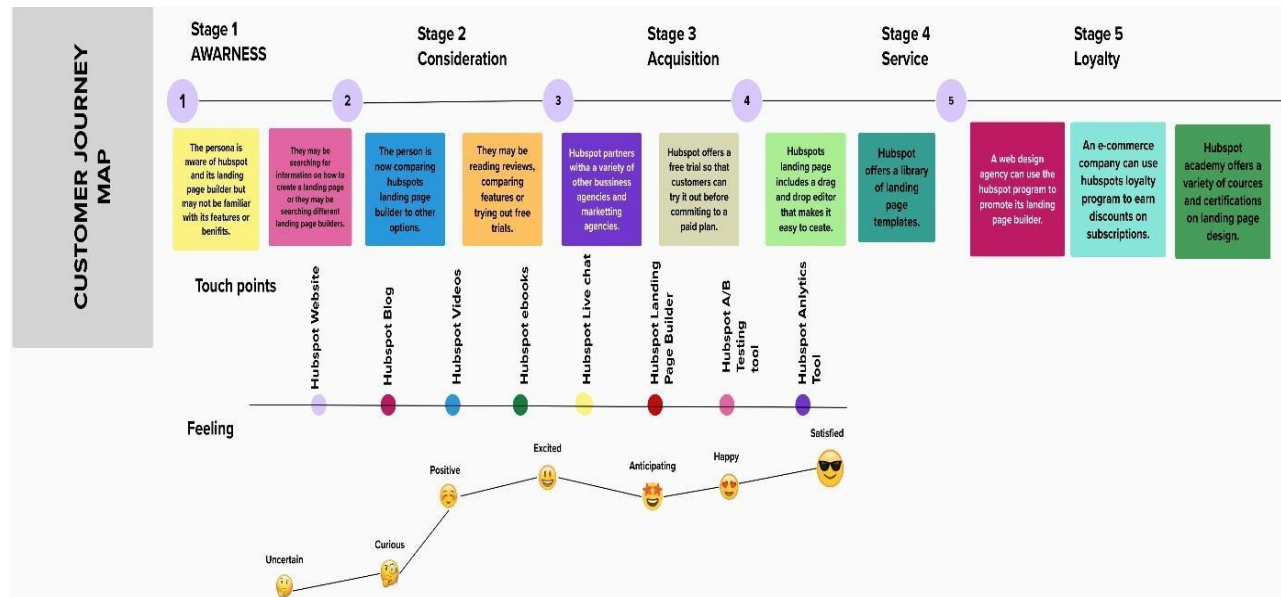


Figure: Architecture of Landing Page of Hubspot

## Project Design Phase-Part 2 :

### Customer Journey Map:



## Requirement Analysis:

### A. Funtional Requirements:

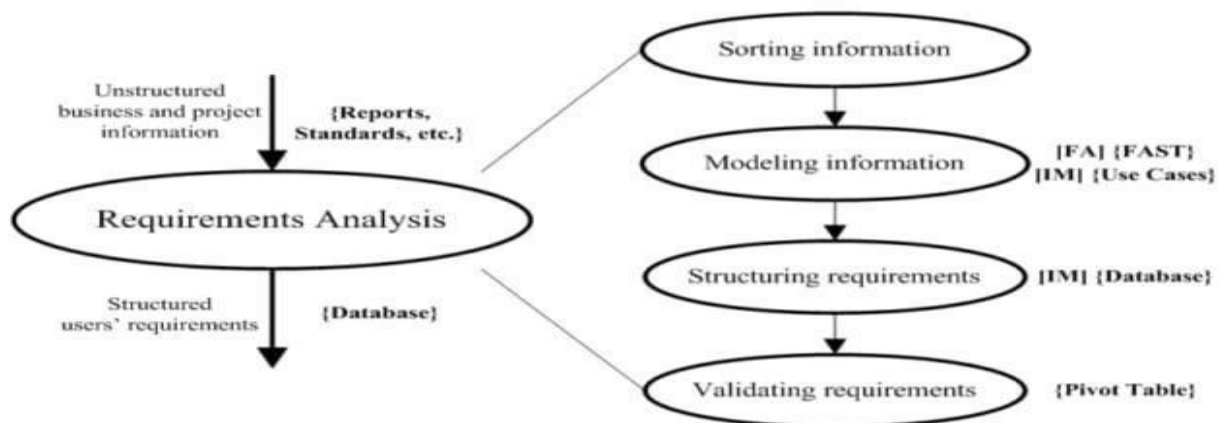
| FR No. | Functional Requirements | Sub Requirement (Story/Sub-Task) |
|--------|-------------------------|----------------------------------|
| FR-1   | Project Features        | Project Properties               |
| FR-2   | Component               | Performance                      |

### B. Non-Functional Requirements:

| FR No. | Non-Funtional Requirements | Description  |
|--------|----------------------------|--|
| NFR-1  | Performance                | Degree a solution performs its designated functions with minimum consumption of resources. |

|       |             |   |
|-------|-------------|---|
| NRF-2 | Scalability | Degree with which a solution can grow and evolve to handle increased amounts of work.                                     |
| NFR-3 | Reliability | Ability of a solution to perform its required functions under stated conditions for a specified period.                   |
| NFR-4 | Security    | Aspects of a solution that protect solution content or components from accidental or malicious access or use.             |
| NFR-5 | Privacy     | The extent to which the system is safeguarded against deliberate and intrusive faults from internal and external sources. |

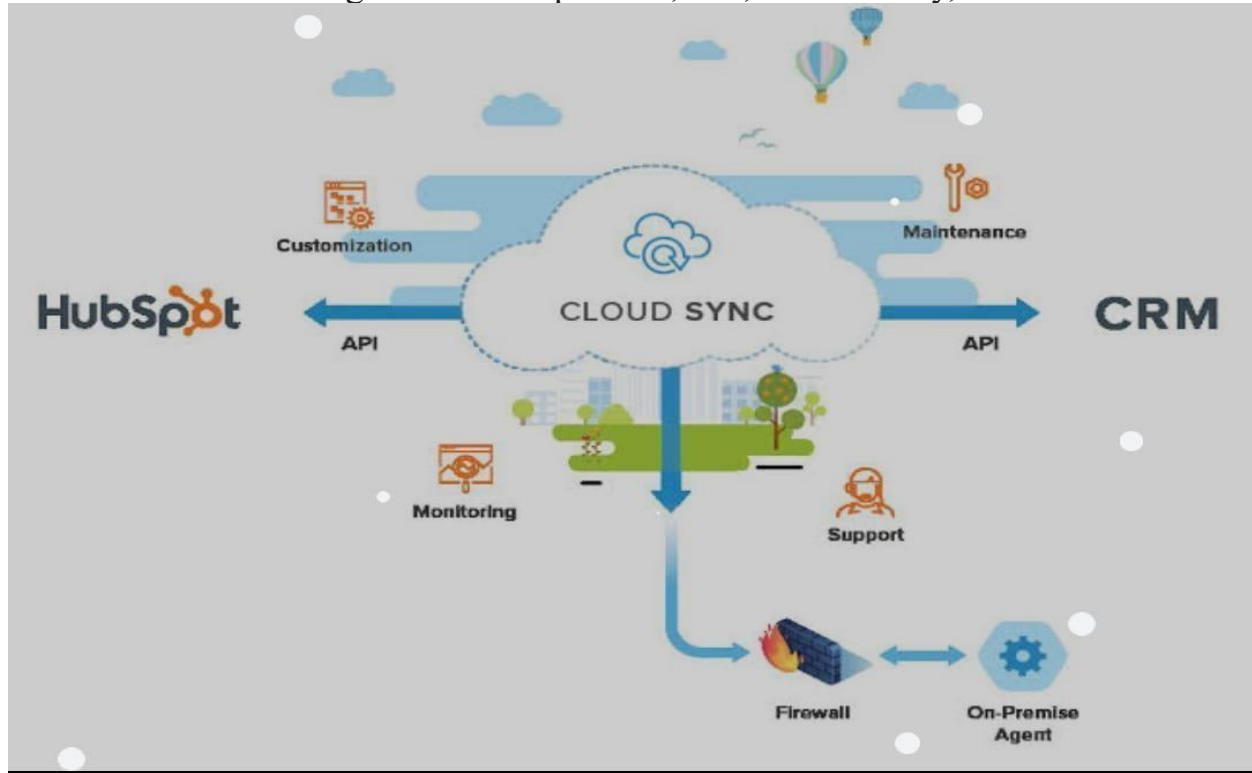
### Flow Charts:



### Technical Architecture:

HubSpot landing pages are built on a scalable and reliable architecture that uses a content management system (CMS), content delivery network (CDN), web application firewall (WAF), load balancer, and web servers. When a visitor requests

a landing page, the request is routed to the load balancer, which distributes it to one of the web servers. The web server retrieves the landing page content from the CDN and delivers it to the visitor. If the visitor submits a form, the form data is submitted to the CMS, which validates and stores it in the CRM. The HubSpot landing page architecture is also designed to be responsive, fast, SEOfriendly, and trackable.

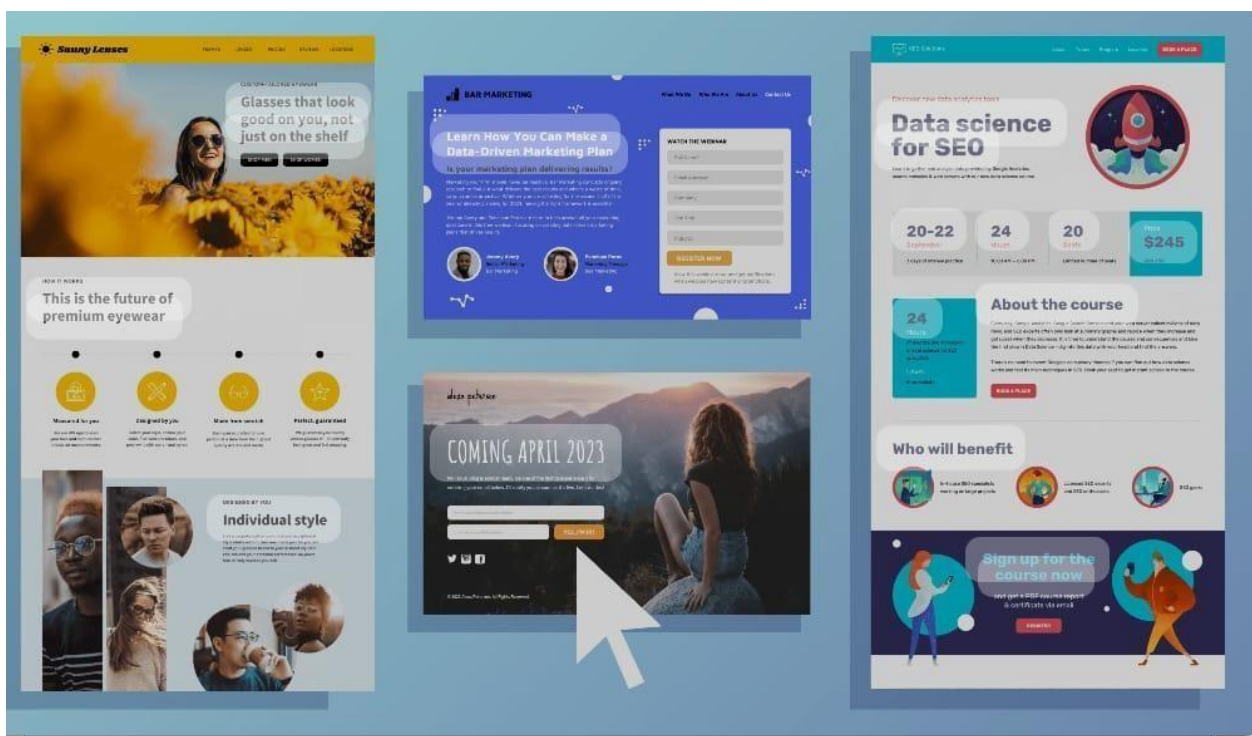


### Open Source Frameworks:

The only open source framework for creating landing pages in HubSpot is CrankShaft Framework. It is a modern framework that accelerates build times on the HubSpot CMS. It is based on a modified version of the popular Eleventy static site generator, and it provides a number of features that make it ideal for building landing pages.

- **CrankShaft Framework:** CrankShaft is a modern framework for accelerating build times on the HubSpot CMS. It is based on a modified Bootstrap 4 framework and includes a number of features that make it ideal for creating landing pages, such as pre-built modules, components, and templates.
- **HubSpot CMS Boilerplate:** The HubSpot CMS Boilerplate is a straight-forward starting point for building a great website on the HubSpot CMS. It is designed to work with both local development and the HubSpot Design Tools.

- **VS Code HubL Language Extension:** The VS Code HubL Language Extension enables super fast local development of CMS pages and is a great complement to using the new local HubL server. It contains comprehensive HubL tag, function, filter, and expression test auto-complete snippets, as well as their documentation.
- **Developer Chrome Extension:** The Developer Chrome Extension is a Chrome/Chromium extension for HubSpot CMS Developers that adds a developer menu, dark theme, and useful shortcuts to commonly used HubSpot query parameters, resources, and tools for making HubSpot Development easier and more enjoyable.



### Third Party API's:

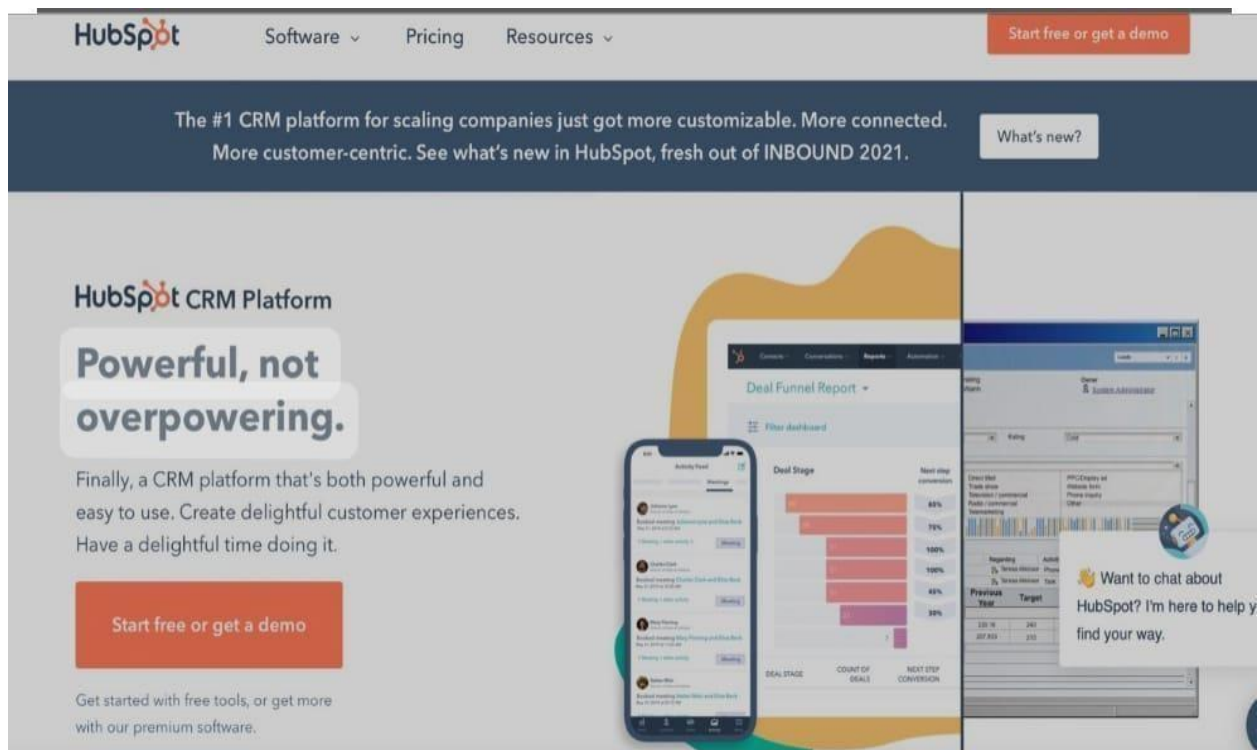
Unbounce is a landing page builder that offers a number of features that make it easy to create high-converting landing pages, such as drag-and-drop editing, A/B testing, and dynamic content personalization. Unbounce also offers an integration with HubSpot that allows you to easily publish your landing pages to HubSpot.

Instapage is another popular landing page builder that offers a similar set of features to Unbounce. Instapage also offers an integration with HubSpot, as well as integrations with other popular marketing automation platforms.

Leadpages is a third-landing page builder that offers a variety of features for creating high-converting landing pages, such as landing page templates, drag-and-drop editing, and A/B testing. Leadpages also offers an integration with HubSpot that allows you to easily publish your landing pages to HubSpot.

HubSpot Landing Pages API is an API that allows you to create, edit, and publish landing pages in HubSpot. The HubSpot Landing Pages API is a good option for developers who want to build custom landing page solutions or integrate landing pages with other systems.

In addition to these dedicated landing page builders, there are a number of other third-party APIs that can be used to create landing pages in HubSpot. For example, the HubSpot Forms API can be used to create custom forms for your landing pages, and the HubSpot CRM API can be used to integrate your landing pages with your CRM system.

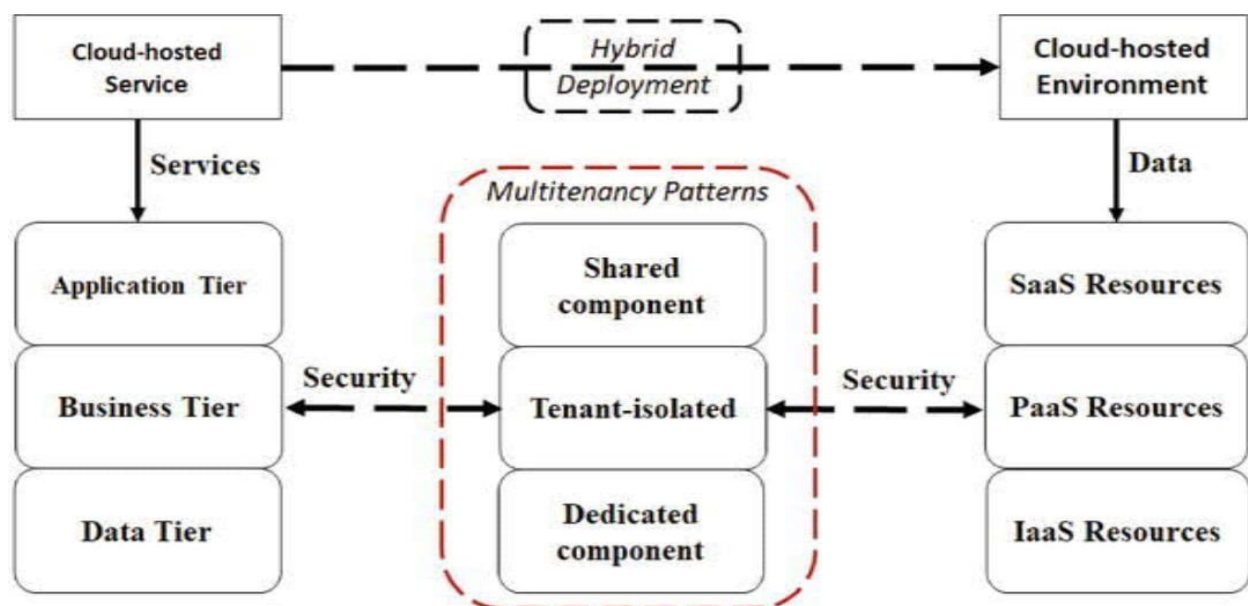


The image is a screenshot of the HubSpot CRM Platform landing page. At the top, there is a navigation bar with the HubSpot logo, links for 'Software', 'Pricing', and 'Resources', and a 'Start free or get a demo' button. Below the navigation bar is a dark blue banner with the text: 'The #1 CRM platform for scaling companies just got more customizable. More connected. More customer-centric. See what's new in HubSpot, fresh out of INBOUND 2021.' and a 'What's new?' button. The main content area features the HubSpot CRM Platform logo and the headline 'Powerful, not overpowering.' Below this, it says 'Finally, a CRM platform that's both powerful and easy to use. Create delightful customer experiences. Have a delightful time doing it.' and another 'Start free or get a demo' button. At the bottom, it says 'Get started with free tools, or get more with our premium software.' The right side of the page shows a collage of HubSpot interface elements, including a 'Deal Funnel Report' with a bar chart showing deal stages (New, Open, Closed) and their counts, a 'Contact List' table, and a 'Reporting' section with a bar chart. A chat bubble on the right says 'Want to chat about HubSpot? I'm here to help you find your way.'



## Cloud Deployment:

1. HubSpot landing pages are automatically hosted in the cloud, so you don't need to worry about setting up or managing your own web server. This makes it easy to get your landing pages up and running quickly and easily, and it also means that you can scale your landing page traffic without having to worry about your server capacity.
2. Use a content delivery network (CDN) to improve the performance of your landing pages. A CDN is a network of servers that are distributed around the world. When a visitor requests your landing page, the CDN will deliver the page from the server that is closest to the visitor. This can help to improve the loading speed of your landing pages, which can lead to higher conversion rates.
3. Cloud-hosted landing pages can scale to handle any amount of traffic. This is important because you never know when a landing page is going to go viral.
4. Cloud-hosted landing pages are very reliable. This is because cloud providers have redundant systems in place to ensure that your landing pages are always up and running.
5. Use a domain name system (DNS) provider to manage your domain names. A DNS provider is a company that translates domain names into IP addresses. When a visitor types in your domain name in their web browser, the DNS provider will direct them to the correct IP address for your landing page.



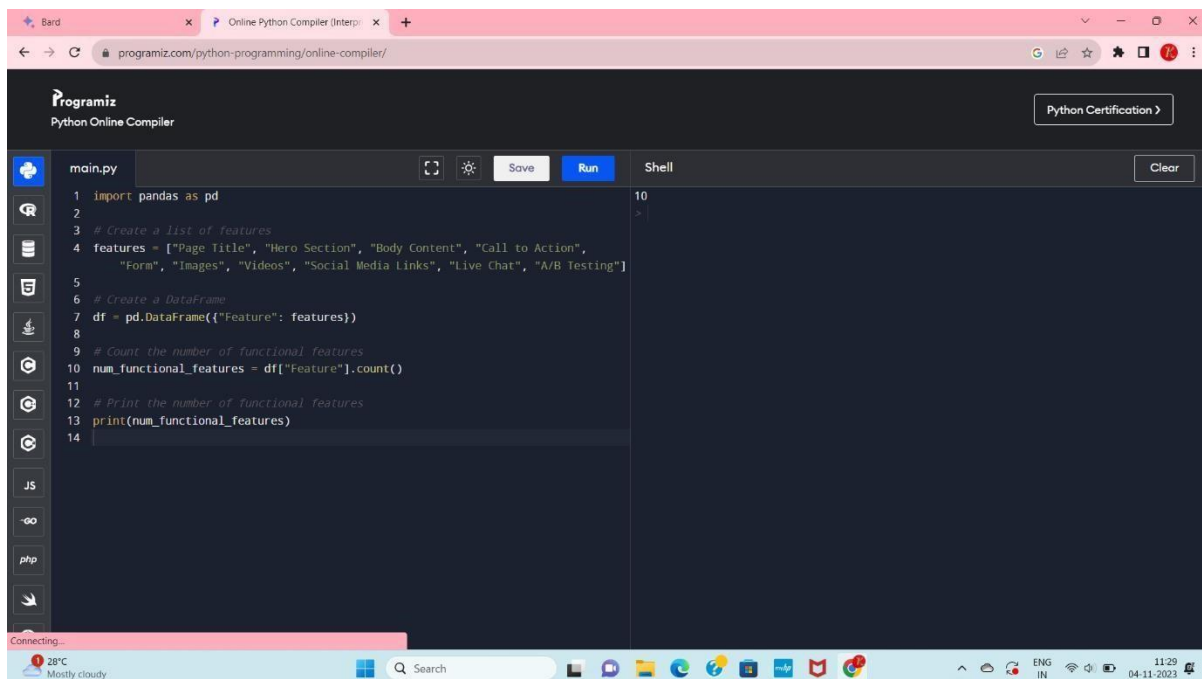
## Project Development Phase:

**Number of Functional Features included in the solution :**

**Python Code:**

```
import pandas as pd #
Create a list of features
features = ["Hero Section", "About Section", "Services Section", "Testimonials
Section", "Call to Action Section"]
# Create a DataFrame df =
pd.DataFrame({"Feature": features}) # Count
the number of functional features
num_functional_features =
df["Feature"].count()
# Print the number of functional features print(num_functional_features)
```

**Output:**



The screenshot shows a web browser window with the URL `programiz.com/python-programming/online-compiler/`. The page title is "Programiz Python Online Compiler". The code editor displays the following Python code:

```
1 import pandas as pd
2
3 # Create a list of features
4 features = ["Page Title", "Hero Section", "Body Content", "Call to Action",
5             "Form", "Images", "Videos", "Social Media Links", "Live Chat", "A/B Testing"]
6
7 # Create a DataFrame
8 df = pd.DataFrame({"Feature": features})
9
10 # Count the number of functional features
11 num_functional_features = df["Feature"].count()
12
13 # Print the number of functional features
14 print(num_functional_features)
```

The output of the code is displayed in the "Shell" window on the right, showing the number 10.



## **Code-Layout, Readability and Reusability :**

### **Code-layout:**

#### **HTML Code:**

```
<!DOCTYPE html>

<html>

<head>

<title>New version</title>

<meta charset="utf-8">

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

</head>

<body>

<header>

<h1>New Version</h1>

</header>

<main>

<section id="hero">



<p>This is the main content section of my landing page.</p>

</section>

<section id="benefits">

<h2>Benefits</h2>

<ul>

<li>Benefit 1</li>
```

<li>Benefit 2</li>

<li>Benefit 3</li>

</ul>

</section>

<section id="cta">

<h2>Call to Action</h2>

<form action="/submit-form" method="post">

<input type="name" name="name" placeholder="Your name">

<input type="email" name="email" placeholder="Your email address">

<button type="submit">Submit</button>

</form>

</section>

</main>

<footer>

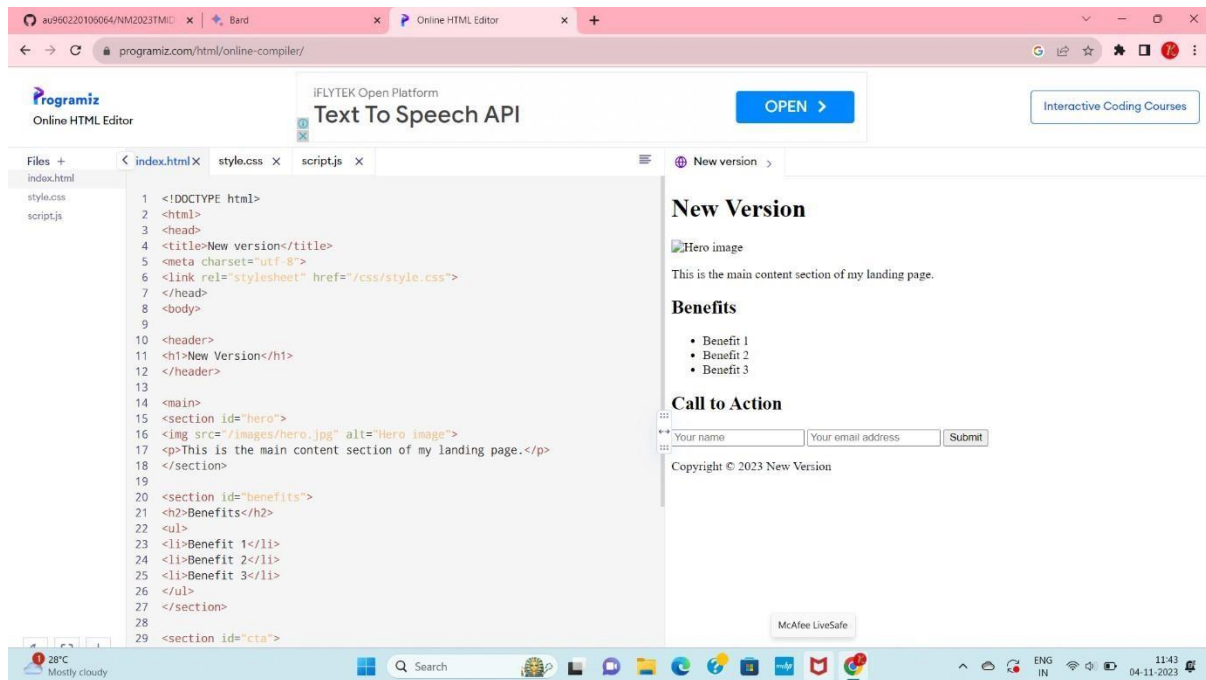
<p>Copyright &copy; 2023 New Version</p>

</footer>

</body>

</html>

## Output:



**Utilization of Algorithms, Dynamic Programming, Optimal memory utilization :**

**Utilization of Algorithms:**

**HTML Code:**

`<!DOCTYPE html>`

`<html>`

`<head>`

`<title>My Landing Page</title>`

`<meta charset="utf-8">`

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

```
<script type="text/javascript"
src="https://js.hsscripts.com/6000000/hs.js"></script>
</head>

<body>
<header>

<h1>My Landing Page</h1>

</header>

<main>

<section id="hero">

<h2>Headline 1</h2>



<p>This is the main content section of my landing page.</p>

</section>

<section id="benefits">

<h2>Benefits</h2>

<ul>

<li>Benefit 1</li>

<li>Benefit 2</li>

<li>Benefit 3</li>

</ul>

</section>

<section id="cta">
```

<h2>Call to Action</h2>

<form action="/submit-form" method="post">

<input type="text" name="userFullName" placeholder="Your name"> <input  
type="email" name="userEmailAddress" placeholder="Your email address">

<button type="submit">Submit</button>

</form>

<script type="text/javascript">

// This script will split the traffic between the two versions of the

headline hbspt.forms.create({ portalId: 'YOUR\_PORTAL\_ID', formId:

'YOUR\_FORM\_ID',

target: '#cta form', variations:

{

'headline': {

'control': 'Headline 1',

'variant': 'Headline 2'

}

}

});

</script>

</section>

</main>

<footer>

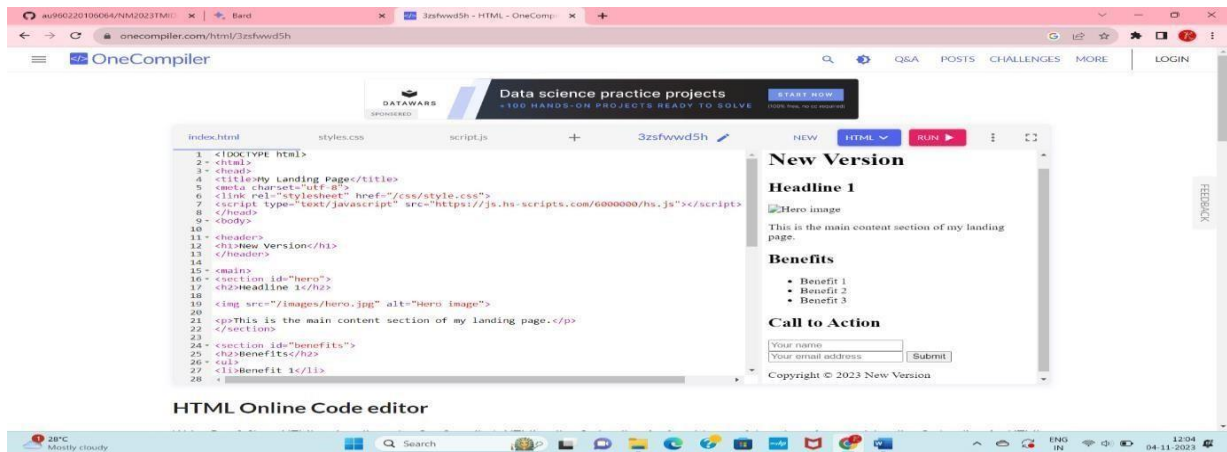
<p>Copyright &copy; 2023 My Company</p>

</footer>

</body>

</html>

**Output:**



**Dynamic Programming:**

**HTML Code:**

<!DOCTYPE html>

<html>

<head>

<title>My Landing Page</title>

<meta charset="utf-8">

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

<script type="text/javascript"  
src="https://js.hs-scripts.com/6000000/hs.js"></script>

</head>

<body> <header>

```

<h1>My Landing Page</h1>

</header>

<main>

<section id="hero">
  {{ module 'hero' }}
</section>

<section id="benefits">
  {{ module 'benefits' }}
</section>

<section id="cta">
  {{ module 'cta' }}
</section>

</main>

<footer>

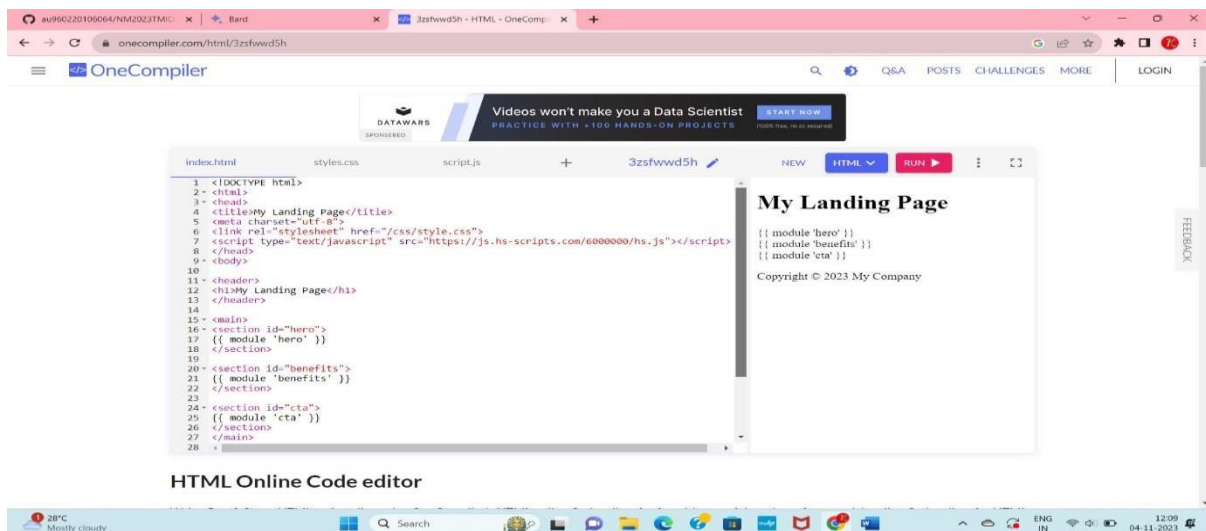
<p>Copyright &copy; 2023 My Company</p>

</footer>

</body> </html>

```

## Output:



**The hero module might contain the following code:**

```
<section id="hero">  
<h2>Headline</h2>  
  
<p>This is the main content section of my landing page.</p>  
</section>
```

**The benefits module might contain the following code:**

```
<section id="benefits">  
<h2>Benefits</h2>  
<ul>  
<li>Benefit 1</li>  
<li>Benefit 2</li>  
<li>Benefit 3</li>  
</ul>  
</section>
```

**Optimal memory Utilization:**

**Code:**

```
{% module 'hero' %}  
<h2>Headline</h2>  
  
<p>This is the main content section of my landing page.</p>  
{% endmodule %}
```



## Debugging and Traceability:

### Debugging and Traceability:

#### HTML Code:

```
<!DOCTYPE html>

<html>

<head>

<title>Your Landing Page</title>

<!-- HubSpot tracking code -->

<!-- Include CSS -->

<link rel="stylesheet" type="text/css" href="your_styles.css">

</head>

<body>

<header>

<!-- Your header content -->

</header>

<main>

<!-- Your main content -->

<h1>Welcome to Our Landing Page</h1>

<!-- Insert dynamic content with HubL -->

<p>{{ content.body }}</p>

</main>

<footer>

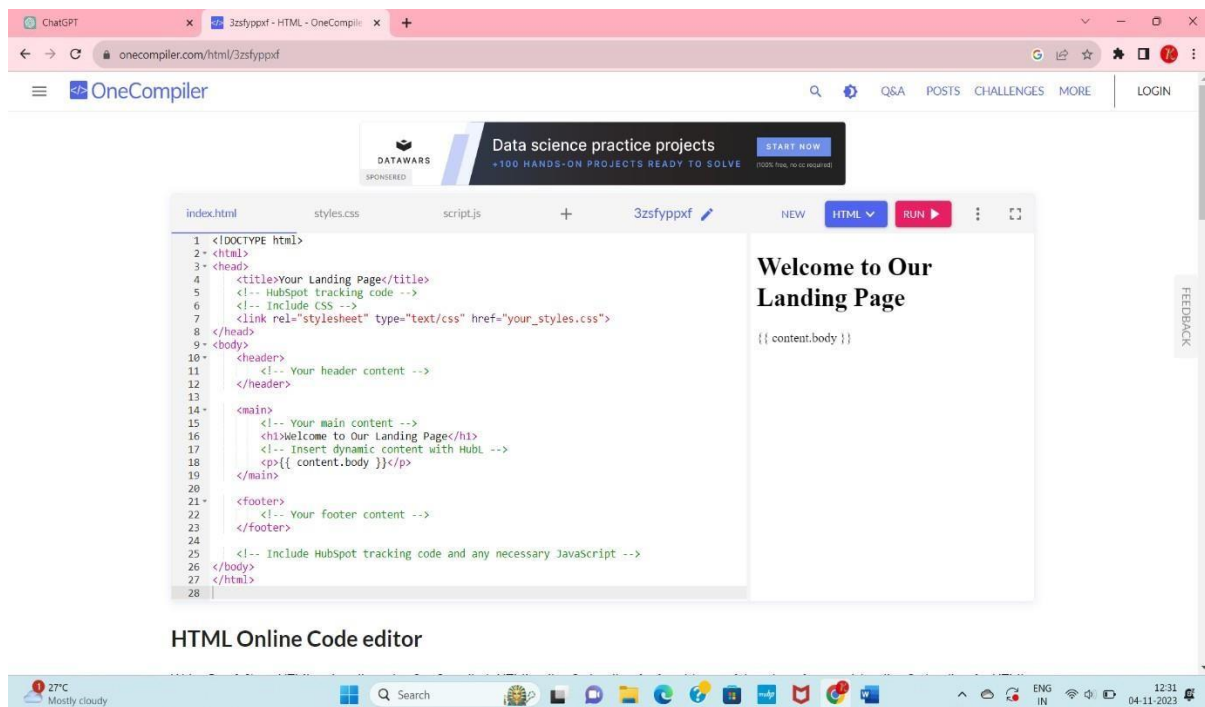
<!-- Your footer content -->
```

</footer>

<!-- Include HubSpot tracking code and any necessary JavaScript -->

</body></html>

## Output:



## Tracking Code:

<form>

<!-- Your form fields -->

<input type="text" name="email" placeholder="Enter your email" />

<button type="submit" onclick="trackFormSubmit()">Submit</button>

</form> <script>     function

trackFormSubmit() {

  \_hsq.push(['trackEvent', id: 'Form Submitted',

  value: 'Landing Page Form Submission'  }]);

```
        // Additional actions after form submission

    }

</script>
```

## Exception Handling:

### Program:

```
import requests

def create_landing_page(title, content, template_id):
    """Creates a new landing page in HubSpot.

    Args:
        title (str): The title of the landing page.
        content (str): The content of the landing page.
        template_id (int): The ID of the landing page template.

    Returns:
        dict:
            A dictionary containing the landing page details, or None if the request fails.

    try:
        response = requests.post(
            "https://api.hubspot.com/landing-pages/v1/pages",
            headers={
                "Authorization": "Bearer YOUR_HUBSPOT_API_KEY",
            },
            json={
                "title": title,
                "content": content,
                "templateId": template_id,
            },
        )

        if response.status_code == 201:
            return response.json()
        else:
            raise Exception(f'Failed to create landing page: {response.status_code}')
```

```

except Exception as e:    print(e)
    return None

def handle_exception(e):
    """Handles an exception that occurred while creating a landing page in HubSpot.

    Args:
        e (Exception): The exception that
        occurred.
    """

    if (e,
        isinstance(requests.exceptions.RequestException):
        print( elif "Failed to connect to HubSpot )
        isinstance(e, API."
        Exception):
            print(e)
    else:
        print("Unknown error occurred.")

if __name__ == "__main__":
    title = "My Landing Page"
    content = "This is the content of my landing page."
    template_id = 123456

    landing_page = create_landing_page(title, content, template_id)

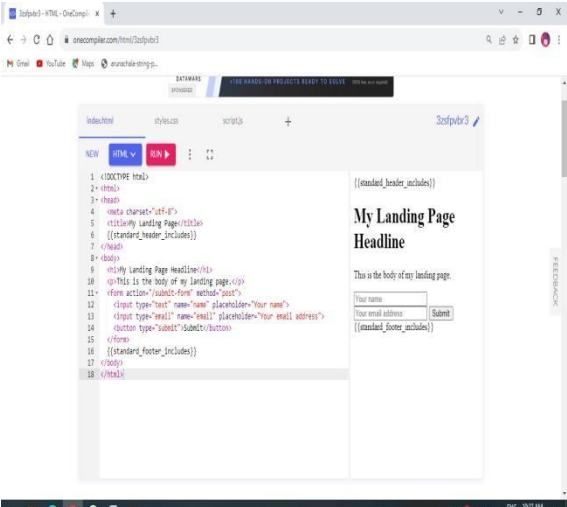
    if landing_page is not None:
        print("Landing page created successfully.")    else:
        handle_exception(e)

```

## Performance & Final Submission Phase:

**Model Performance Metrics:**

**Model Performance Testing:**

| Parameter | Values                                     | Screenshot   |
|-----------|--|--|
| Metrics   | Online HTML Compiler and Output Screenshot |  <p>The screenshot shows a web browser window displaying an online HTML compiler. The compiler interface includes a code editor on the left with the following HTML code:</p> <pre> 1 &lt;!DOCTYPE html&gt; 2 &lt;html&gt; 3 &lt;head&gt; 4   &lt;meta charset="utf-8"&gt; 5   &lt;title&gt;My Landing Page&lt;/title&gt; 6   {{standard_header_includes}} 7 &lt;/head&gt; 8 &lt;body&gt; 9   &lt;h1&gt;My Landing Page Headline&lt;/h1&gt; 10   &lt;p&gt;This is the body of my landing page.&lt;/p&gt; 11   &lt;form action="/submit-form" method="post"&gt; 12     &lt;input type="text" name="name" placeholder="Your name"&gt; 13     &lt;input type="email" name="email" placeholder="Your email address"&gt; 14     &lt;button type="submit" value="Submit"&gt;Submit&lt;/button&gt; 15   &lt;/form&gt; 16   {{standard_footer_includes}} 17 &lt;/body&gt; 18 &lt;/html&gt; </pre> <p>On the right side of the compiler, the rendered output is shown. It features a heading "My Landing Page Headline" and a form with two input fields labeled "Your name" and "Your email address", followed by a "Submit" button. The browser's address bar shows the URL "https://onecompiler.com/html/1zfp4b3".</p> |

### Demo Link:

[https://drive.google.com/file/d/1f\\_gb7bq8tlbn87HFvcGjjQcC9Zzd4Rs/view?usp=drivesdk](https://drive.google.com/file/d/1f_gb7bq8tlbn87HFvcGjjQcC9Zzd4Rs/view?usp=drivesdk)