**Exercise 1: Exploring the Contoso Conference Application Scenario In this exercise, you will run the Contoso Conference web application and examine each of the functions it provides**.

* The Contoso Conference web application contains the following pages:
* The Home page, which provides a brief overview of the conference, the speakers, and the sponsors. The Home page also includes a video from the previous conference.
* The About page, which provides more detail about the conference and the technologies that it covers.
* The Schedule page, which lists the conference sessions. The conference has two concurrent tracks, and the sessions are organized by track. Some sessions are common to both tracks.
* The Register page, which enables the user to provide their details and register for the conference.
* The Location page, which provides information about the conference location and a map of the venue.
* The Live page, which enables an attendee to submit technical questions to the speakers running the conference sessions. The page displays the answer from the speaker, together with questions (with answers) posted by other conference attendees.
* The Feedback page, which enables the user to rate conference sessions and speakers.

The main tasks for this exercise are as follows:

1. Start the web application and view the Home page.
2. View the About and Schedule pages.
3. View the Register page and register as a new attendee.
4. View the Location page.
5. Submit a question and provide conference feedback.

Start Visual Studio.

1. Open the ContosoConf solution in the folder
2. Start the application without debugging.

The Home page is displayed in Internet Explorer, like this:

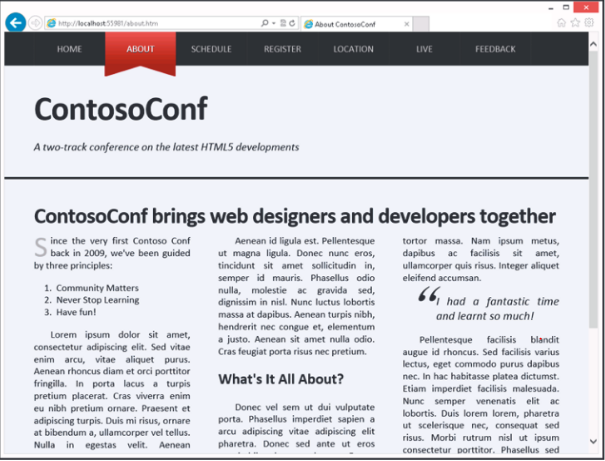


The graphics elements for the speakers and sponsors are implemented by using HTML img elements. The sources of the images are jpg and png files.

1. Scroll to the bottom of the page and play the video from the previous conference. This functionality is implemented by using the HTML5 video element.
2. Pause the video
3. Scroll to the top of the Home page and hover the mouse over the Register Free icon. Notice that the icon rotates and expands as the mouse enters the icon. This feature is implemented by using CSS.
4. At the very top of the page, move the mouse over the menu bar listing the names of the pages in the application. Do not click any menu items. Notice that each item is highlighted as the mouse traverses it. This feature is implemented by using an HTML nav element and CSS.

**Task 2: View the About and Schedule pages**

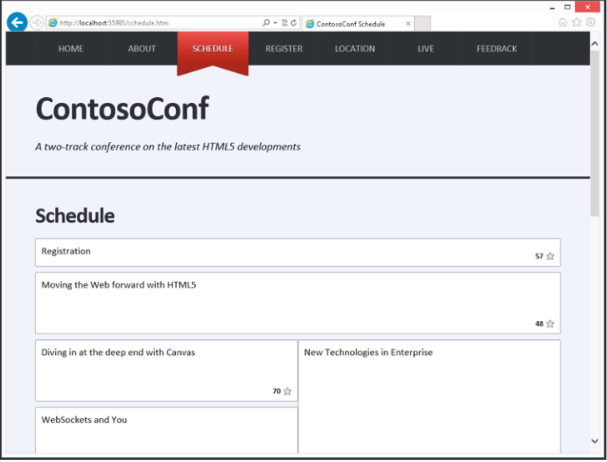
1. Using the menu bar, move to the About page. The About page looks like this:



Notice that when you click an item in the menu bar, the style of the item changes; it is displayed with a ribbon effect. This feature is implemented by using CSS.

The other styling features, including the large drop-capital "S" at the start of the first paragraph, the column layout, and the quotation in the third column, are also implemented by using CSS

1. Move to the Schedule page. The Schedule page looks like this:



The list of sessions is held in a database that is accessed by using a web service. This page uses JavaScript code to connect to the web service, retrieve the list of sessions, and dynamically populate the body of this page with the session information.

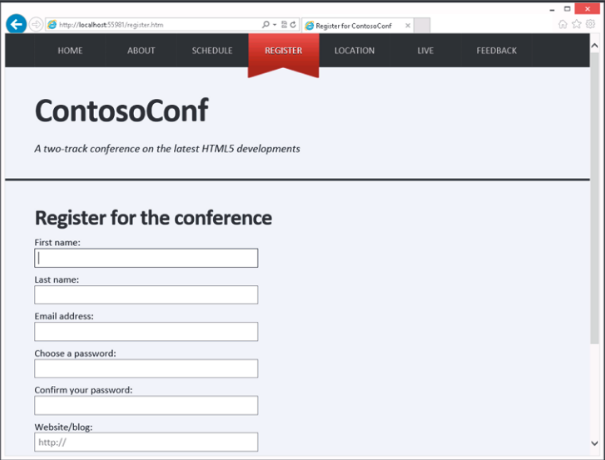
1. Select the session Moving the Web forward with HTML5 and click the star icon. When this happens, notice that the star changes color and that number next to the star increases. This number indicates how many attendees have expressed an interest in this session; to get a good seat, the user may need to arrive early for popular sessions.
2. Click the star again to deselect the session. The number of interested attendees drops by one.

The functionality is implemented by a combination of CSS and JavaScript code that sends information to another web service about the sessions that a user selects.

**Task 3: View the Register page and register as a new attendee.**

1. Move to the Register page.

The Register page looks like this:



1. Register the details for a new attendee. Enter the following information and then click Register

* First name: Erico
* Last name: Grubero
* Email address: dummy data

Notice that the page performs the following validations:

* All fields apart from Website/blog are mandatory.
* The Email address must be in the correct format.
* The password must contain at least 5 letters and numbers.
* The value entered for the Confirm your password field must match the password field. This validation is performed by using a combination of HTML5 forms validation controls, and JavaScript code. The styling of the fields when they display an error is controlled by using CSS.

This validation is performed by using a combination of HTML5 forms validation controls, and JavaScript code. The styling of the fields when they display an error is controlled by using CSS.

1. Complete the data by providing the following information, and then click Register:

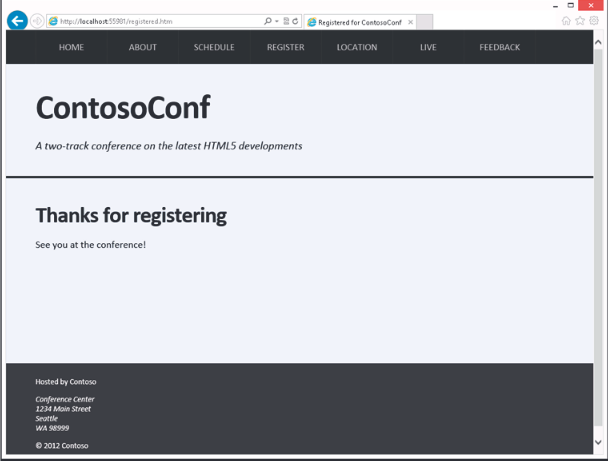
* Email address: [grubere@contoso.com](mailto:grubere@contoso.com)
* Choose a password: abc1234
* Confirm your password: wxyz9999

Notice that this time a different error message appears because the values specified for the two password fields are not the same.

1. Change the value in the Confirm your password field to abc1234, and then click Register again.

When you have successfully registered, the confirmation page appears.

The confirmation page looks like this:



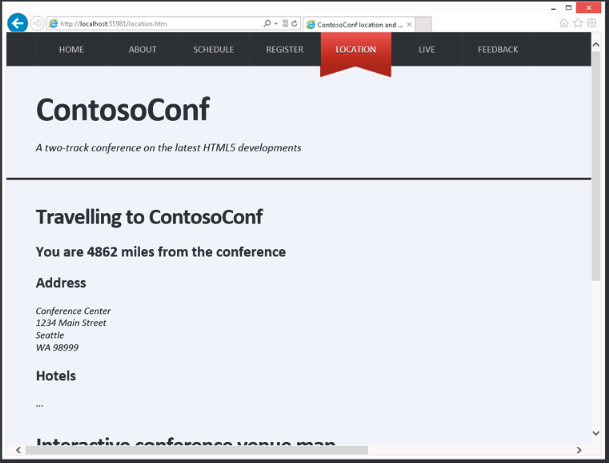
**Task 4: View the Location page**

1. Move to the Location page.

If the message localhost wants to track your physical location appears in the Internet Explorer message bar, click Allow once. In the Enable Location Services message box, click Yes.

The page displays information about your current location (the distance from the conference venue) by using the Geolocation API in JavaScript.

The Location page looks like this:

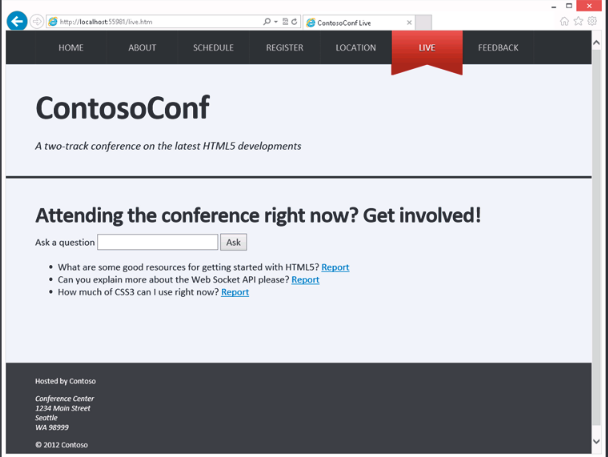


1. Scroll to the bottom of the page. The venue map that appears is generated by using Scalable Vector Graphics.

**Task 5: Submit a question and provide conference feedback**

1. Move to the Live page.

The Live page looks like this:



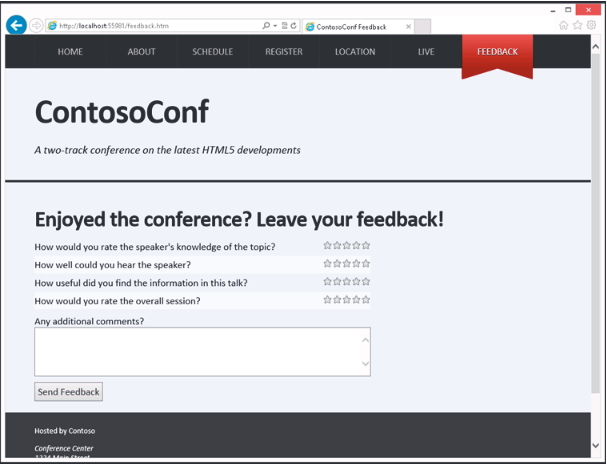
The Live page enables an attendee to submit questions to a speaker and to view the response. The page also displays questions asked by other attendees.

1. Type the question What is the best way to learn HTML5?, and then click Ask.
2. Review the questions that are displayed. This page also enables an attendee to report any questions that they feel are unsuitable or offensive
3. Select the question that you just asked and report it. The question will be vetted and then disappear.

Questions and reporting are managed by using a web socket server. The application connects to this server by opening a client connection and sending requests asynchronously. As other attendees post questions, the JavaScript code behind this page automatically updates the list of questions that is displayed.

1. Move to the Feedback page.

The Feedback page looks like this:



The Feedback page enables an attendee to provide feedback about a session by selecting a star rating and by providing additional comments.

The star rating is implemented by using a combination of JavaScript code and CSS styles behind HTML5 input fields.

1. Provide a rating for a session; click the third star adjacent to the first question, the fifth star adjacent to the second question, and the fourth star adjacent to the two remaining questions.
2. In the Any additional comments box, type Good conference, and then click Send Feedback.

Notice that when you send the feedback, the form flies off the screen to indicate that the feedback has been posted. This animation is performed by using CSS.

1. Close web Explorer.

*Results: After completing this exercise, you will be able to describe the features of the Contoso Conference web application and list the technologies that are used to implement them*.

**Exercise 2: Examining and Modifying the Contoso Conference Application**

Scenario

In this exercise, you will examine the Visual Studio project for the Contoso Conference application. You will see how the project is structured, and how the files and scripts for the project are organized into folders. You will then run the application again, make some modifications to the HTML markup and CSS, and view the results.

The main tasks for this exercise are as follows:

1. Explore the web pages for the application by using Visual Studio.
2. Explore the structure of the project.
3. Run the application and make live modifications.

**Task 1: Explore the web pages for the application by using Visual Studio**

1. In Visual Studio, open the index.htm file. This file contains the HTML markup for the Home page as static text. Examine the following items in the file:

* The nav element at the start of the body section. This element defines the menu that appears at the top of the page (the same menu appears on the other HTML pages as well). The item tagged with the active class specifies the item that refers to the current page. This item is styled differently when it is rendered.
* The section with the video class above the page footer. This section implements the video player.
* The link elements near the top of the file. These elements specify the CSS files that provide the styling for this page. The index.css style sheet contains the styles specifically for this page, while the other style sheets contain styles that are used throughout the application.
* The script elements just before the closing body tag. These elements specify the JavaScript files that implement the functionality for this page.

1. Open the about.htm file. This file contains the HTML markup for the About page as static text.

Notice that:

* This page implements the same navigation menu as the Home page. Notice that the About item is tagged with the active class; this causes the About item to be displayed using the ribbon style when it is rendered by using the nav.css stylesheet.
* Styling is handled by a set of CSS files. The about.css style sheet implements the styling specific to this page.

1. Open the schedule.htm file. This file contains the HTML markup for the Schedule page.

In this page, notice that the list of sessions in the <section class="page-section schedule> element is empty; it is populated when the page is displayed by using the JavaScript code in the schedule.jsscript referenced near the end of the file.

1. Open the register.htm file. This file contains an HTML form in the <section class="page-section register"> element. This form validates the data that an attendee enters.

When the user submits the form, their details are posted to the registration service at the URL registration/new.

1. Open the location.htm file. This file contains an HTML page that displays the distance of the user from the conference site, together with a venue map.

The distance to the conference site is calculated by using JavaScript code that calls the Geolocation API, in the script location.js. The script displays the distance in the <h2> element with the id of distance in the <section class="travel"> element. The venue map is drawn by using Scalable Vector Graphics in the <section class="venue"> element.

1. View the live.htm file.

This file contains a form in the <section class="page-section Live"> element that enables a user to submit questions.

Questions are posted to a server listening on a web socket.

Questions posted by other users are received by using a web socket, and then added to the list on the page. The JavaScript code that implements the web socket code is located in the live.js file.

1. View the feedback.htm file. This page contains the feedback form in the <section class="page-section feedback"> element, enabling attendees to provide their feedback on the conference.

The input fields for the first four questions are rendered as stars by using the JavaScript code in the StartRatingView.js file and the styles in the feedback.css style sheet. Properties of the input fields define the maximum and minimum ratings, and each rating is displayed as a single yellow star. The input field for the comments feedback is a <textarea> element. When the user submits the feedback, JavaScript code in the feedback.js file and styles in the feedback.css style sheet animate the form to make it fly off the screen.

**Task 2: Explore the structure of the project**

The files for the project are organized into the following folders. In Solution Explorer, examine the contents of each folder in turn:

**images**. This folder contains photographs of the conference speakers, and logos of conference sponsors.

**scripts**. This folder contains the JavaScript files used throughout the application. The **pages** subfolder contains the JavaScript files containing the code that is specific to each page. Each file is named after the corresponding HTML file.

**styles**. This folder contains the styles for the application. It is organized in a similar manner to the scripts folder. The **images** subfolder contains the graphic image of a star, used by the feedback and schedule pages.

**Note**: For the purposes of this lab you can ignore the Controllers and Views folders. These folders contain C# and ASP.NET code that implement the web services used by the application. In the real world, they would be implemented separately from the web application. You can also disregard the Properties and References folders, which contain items that support the web services, as does the Global.asax file. You will not use any of these items in this course.

**Task 3: Run the application and make live modifications**

1. Build and run the web application without debugging, and display the Home page.
2. Leave the application running and return to Visual Studio.
3. Edit the HTML markup for the Home page and change the text for the Register Free button to Register Now.
4. Open the nav.css style sheet in the styles folder; this style sheet contains the styles used to render the contents of the <nav> element and change the background color to blue.
5. Save the changes, return to Internet Explorer, refresh the view, and verify that you can see the effects of the changes.

*Results: After completing this exercise, you will be able to describe how the Contoso Conference application is structured as a Visual Studio project.*

Lab 2: Creating and Styling HTML5 Pages

**Scenario**You are a web developer working for an organization that builds websites to support conferences. You  
have been asked to create a website for ContosoConf, a conference that showcases the latest tools and  
techniques for building HTML5 web applications.  
You decide to start by building a prototype website consisting of a Home page that acts as a landing  
page for conference attendees, and an About page that describes the purpose of the conference. In later  
labs you will enhance these pages and add further pages that enable attendees to register for the  
conference, and that provide information about the sessions scheduled to run as part of the conference.

**Objectives**After completing this lab, you will be able to:  
• Create HTML5 pages.  
• Style HTML5 elements.

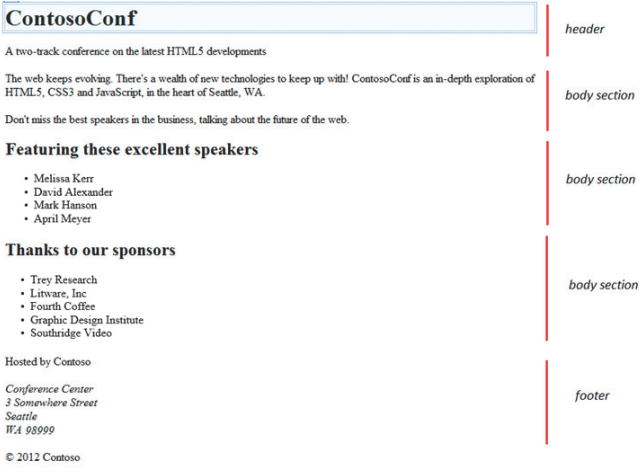
Exercise 1: Creating HTML5 Pages  
**Scenario**In this exercise, you will begin to create the ContosoConf website.  
First you will create a new ASP.NET Web Application. Then you will add two HTML files for the Home and  
About pages. Next, you will add navigation links to the pages. Finally you will run the web application and  
verify that the Home page and About page are formatted correctly.  
The main tasks for this exercise are as follows:  
1. Create a new ASP.NET web application.  
2. Add the Home page.  
3. Add images to the Home Page.  
4. Add the About page.  
5. Add navigation links.  
6. Run the web application.

Task 1: Create a new ASP.NET web application  
1. Start Visual Studio.  
2. Create a new web application named **ContosoConf** in the **E:\Mod02\Labfiles\Starter\Exercise 1**folder. Use the **Visual C# ASP.NET Empty Web Application** template to create this web application.

**Task 2: Add the Home page**  
1. Add a new HTML page named **index.htm** to the ContosoConf project. This page is the default page  
for the website, and will be displayed when a user browses to the URL for the website.  
2. Using Notepad, open the **index.txt** file located in the **E:\Mod02\Labfiles\Starter\Exercise  
1\Resources** folder.  
• Examine this file, and then add HTML5 elements to the index.htm file in the web application that can  
display the items specified in the index.txt file.  
• Use HTML5 elements, such as **<header>**, **<section>**, and **<footer>** where appropriate.  
For example, the following <header> element contains the content from the index.txt file:

<header>  
<h1>ContosoConf</h1>  
<p>A two-track conference on the latest HTML5 developments</p>  
</header>

The following image shows how the unstyled page should look, and how the content should be divided  
into the elements on the page.



**Task 3: Add images to the Home Page**  
1. Add the speaker and sponsor images to the ContosoConf project. These images are located in the  
E:\Mod02\Labfiles\Starter\Exercise 1\Resources folder. They should be added to a new folder called  
**images** in the project.  
2. Update the HTML markup in the **index.htm** file to include the images from the **speakers** and  
**sponsors** folders in the **images** folder. For example:

<img src="/images/speakers/melissa-kerr.jpg" alt="Melissa Kerr"/>  
Melissa Kerr

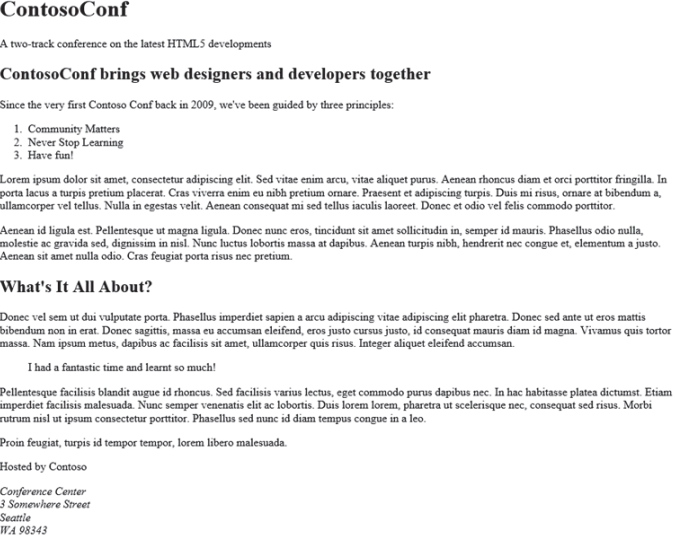
The following image shows how the page should appear:

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

**Note:** In an **<img>** element, you can use the **Pick URL** wizard to specify the **src** attribute for an  
image.

Task 4: Add the About page  
1. Add a new HTML page, named **about.htm** to the ContosoConf project.  
o Use the **Add New Item** command on the **Project** menu.  
o Select the **HTML Page** template.  
2. Add HTML elements to **about.htm,** using the text provided in the **about.txt** file located in the  
folder.  
• Add an appropriate title to the page.  
• Copy the **<header>** and **<footer>** elements from **index.htm**.  
• Use the **<article>**, **<blockquote>**, and**<ol>** elements where appropriate.  
The following image shows how the unstyled page should look.



**Task 5: Add navigation links**1. Add a navigation element to **index.htm** and **about.htm**. The navigation element should contain links  
to both pages.  
For example, the **<nav>** element for the index.htm page should look like this:  
<nav>  
<a href="/index.htm">Home</a>  
<a href="/about.htm">About</a>  
</nav>  
**Task 6: Run the web application**1. Run the web application by using Internet Explorer.  
2. Verify that the correct text and images are displayed.  
3. Verify that the navigation links reference the correct pages.

**Results**: After completing this exercise, you will have built a simple HTML5 web application with a Home  
page and an About page

**Exercise 2: Styling HTML pages**  
**Scenario**In this exercise, you will add styling to the Home and About pages.  
You will create a stylesheet in the ContosoConf project. Then you will add CSS rules to style the Home and  
About pages to match a specified design. Finally, you will run the web application and verify that the  
pages are styled correctly.  
The main tasks for this exercise are as follows:  
1. Create a new style sheet

2. Add CSS rules to style the pages.  
3. Run the web application.

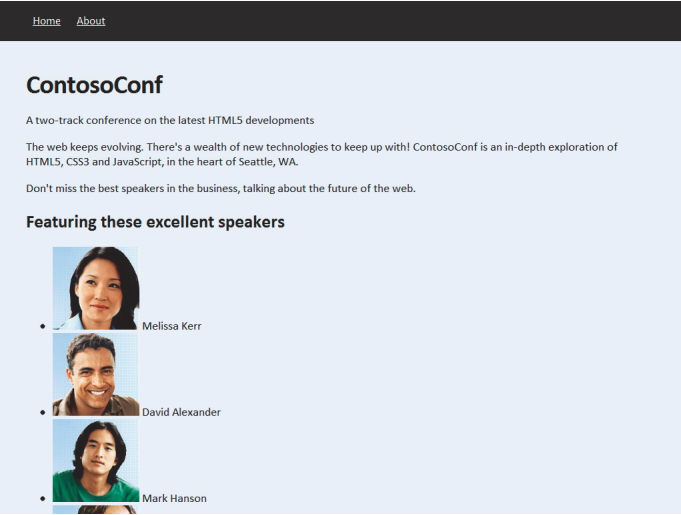
**Task 1: Create a new style sheet**  
1. Add a folder named **styles** to the ContosoConf project.  
2. Add a stylesheet named **site.css** to the **styles** folder.  
o Use the **Add New Item** command on the **Project** menu.  
o Select the **Style Sheet** template.  
3. Add a link to the **site.css** style sheet from the **index.htm** and **about.htm** HTML5 pages.  
o Use a **<link>** element inside the page header; set the **href** property to **"styles/site.css"**, set the  
**rel** property to **"stylesheet"**, and set the **type** property to **"text/css"**.

**Task 2: Add CSS rules to style the pages**  
1. Add a CSS rule to the site.css style sheet to style the **<html>** element of the Home and About pages  
(the same styling rules should apply to both pages):  
o Set **background-color** property for the web page to **#EAEEFA**, set the list of fonts in the font  
family to **Calibri**, **Arial**, **sans-serif**, and set the font size **62.5%**2. Add a CSS rule to style the **<body>** element of a web page:  
o Set the **margin** property to **0**, and the **font-size** property to **1.8rem**.  
3. Add a CSS rule to style the **<nav>** element of a web page:  
o Set the **background-color** property to **#1d1d1d**, set the **line-height** property to **6rem**, and set  
the **font-size** property to **1.7rem**.  
o Additionally, add a style for all links (**<a>** elements) that occur inside a **<nav>** element; set the  
**color** property to **#fff** and the **padding** property to **1rem.**

4. Add a CSS rule to style the **<h1>** element of a web page:  
o Set the **font-size** property to **4rem**, set the **letter-spacing** property to **-1px**, and set the **margin**property to **1em 0 0,25em 0**.  
5. Modify the HTML markup for the Home and About pages and wrap the **<a>** elements in the **<nav>**section in **<div class="container">** element, like this:

<div class="container">  
<a href="/index.htm">Home</a>  
<a href="/about.htm">About</a>  
</div>

Add rule to the site.css style sheet to achieve the horizontally centered, fixed-width column effect for  
all sections marked with the container class.  
o Set the **padding** property to **0.1rem**, the **max-width** property to **94rem**, and the **margin**property to **0 auto**.

The styled Home page should look similar to this:  


Task 3: Run the web application  
1. Run the web application and verify that the About page and Home page are both styled  
appropriately.  
The About page should look like this in Internet Browser:

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente