

資訊專案文件與開源碼詮釋資料中文化
第一項
使用漸進增強策略打造有韌性的前端頁面

執行單位：國家資通安全研究院

中華民國112年9月

使用漸進增強策略打造有韌性的前端頁面

<https://www.gov.uk/service-manual/technology/using-progressive-enhancement>

漸進增強是一種建立網站與應用程式的策略，它的核心概念是：先讓網頁能單純以HTML運作後，再加入其他，如CSS與JavaScript等技術。

因為HTML是最具韌性的前端技術，當HTML出錯時，就沒有網頁；但如果CSS或JavaScript出錯時，HTML仍會正確地呈現。

使用漸進增強意味著，即使HTML網頁中所加入CSS與JavaScript等技術出現問題時，使用者仍可以完成他們所需執行的操作。使用漸進增強建構服務有以下好處：

- 使您的服務更具韌性
- 確保您服務的最基礎功能可以運作並滿足使用者核心需求
- 透過鼓勵最佳實作方式來提高無障礙性，例如：[編寫語意標記](#)
- 讓設備或網路條件不佳的使用者也能使用您的服務

1 從HTML開始

大部分政府網站的服務應該僅使用HTML即可運作，這些包含以下類型服務：

- 交換資料性(transactional)服務，例如：使用者填寫資訊並提交給政府的表單
- [smart answers](#)服務，例如：GOV.UK 的[海外出生登記服務](#)
- 基於內容的網站，例如：GOV.UK 的[國際旅遊建議](#)

2 使用互動性元素(interactive elements)

您可以選擇使用互動性元素(interactive elements)，但必須要有一個相同核心功能的備援方案。例如，讓動態自動完成元素可退回到 `<select>` 或其他類似元素。即便互動元素失效了，使用者仍可以進行他們需要做的事情。

您還必須合理地組織[原始碼\(source\)順序與文件大綱](#)。這種方法會為服務提供堅韌的基礎，同時意味著您的網站將可以與大多數的新舊[設備與瀏覽器](#)相容。此策略有助於確保網站具有無障礙性(accessible)，因為使用者將能透過HTML存取他們所需的一切。

譯註：「設備與瀏覽器」翻譯請參考中文化項目二：針對不同的瀏覽器和裝置進行設計。

3 增添更多層次元素

一旦您的服務已經使用了HTML打造基礎後，您可以開始新增以下內容：

- 圖片
- 風格樣式
- 影音媒體
- 腳本(script)，例如：JavaScript
- 更順暢、快速且不必重新整理整個頁面的互動體驗

- 實際送出資料前預先確認資料有效性的驗證方式
- 互動式圖表

4 創造更複雜的服務

使用漸進增強來建構服務為您提供一個具有韌性的基礎，可以用來建構更複雜的服務。在這種情況下，將服務視為一體考量不見得有幫助，而應將服務拆分為不同的互動模式，並決定使用JavaScript是否提供更好的使用體驗。

某些政府服務的特殊功能無法僅憑HTML實現。例如：「英國環境小組」的[洪水地圖服務](#)，需要使用JavaScript呈現洪水的風險區域。但這樣做會造成一些問題，例如：

- 身心障礙者將無法使用部分的功能
- 您的服務可能因依賴JavaScript而不可靠(reliable)

為了避免這些缺點，首先應嘗試在不使用JavaScript的情況下，網頁功能也能正常運作。若行不通，則需要有一個替代機制取代JavaScript所提供的互動效果。雖然效果可能不如啟用JavaScript的互動體驗好，但使用者也因此仍然可以使用您的服務達成目的。例如，資料可呈現為無障礙性(accessible)的[資料表格](#)。

如果以上方法都行不通，您需證明已考慮到輔助科技的使用者如何使用您的服務，含透過非數位途徑，例如打電話或臨櫃辦理。

4.1 使用JavaScript框架

有些開發者認為使用JavaScript框架開發服務很方便。但某些JavaScript框架會要求使用者下載大量的資料，特別是使用者的網路速度較慢或使用較老舊的設備時，可能會導致效能問題。因此，我們應該權衡使用JavaScript框架的好處，以及是否值得承擔這些潛在的效能問題。

使用JavaScript框架可能導致其他問題，例如：

- 您的開發者依賴於無法掌握的第三方程式碼。
- 當框架的受歡迎程度下降時，則難以找到維護框架的技術資源。

您需要確保JavaScript框架將元素作為獨立的元件載入。如果JavaScript載入失敗，僅會是特定的單一元件失效，網頁其餘部分仍可運作如常。如果您的服務使用JavaScript框架來呈現整個頁面，一旦框架載入發生問題，將會立即導致服務的中斷。這些獨立的元件也需要有備援策略，即使JavaScript失效了，使用者仍然能夠使用服務。

伺服器應該能退回以HTML進行渲染網頁，並照常將原始碼交付至瀏覽器，於必要時再使用JavaScript來增強元件。

4.2 使用JavaScript伺服器端渲染(server-side rendering, SSR)

您可以使用伺服器端渲染(server-side rendering)或稱[Isomorphic JavaScript](#)或[Universal JavaScript](#)。然而，您仍需要在用戶端(client-side)停用JavaScript測試服務是否正常運作。當JavaScript失效時，仍能透過伺服器端渲染HTML來維持服務的正常運作。

5 測試您的服務

您須測試服務中大量依賴JavaScript或JavaScript框架的元件，來確保其無障礙性，由實際(身心障礙)的使用者進行測試更為理想。您也需要測試前端的效能。

6 不要假設使用者關閉CSS或JavaScript

不要只因為預想使用者會選擇停用這些功能，便設計一個無CSS或JavaScript運作的服務，還有許多可能的情境使得這些附加層失效或遭到濾除，例如：

- 短暫性的網路中斷
- 第三方瀏覽器擴充軟體的介入，例如：廣告遮擋軟體等
- 第三方供應商中斷服務，例如：內容傳遞網路(content delivery network, CDN)的服務異常
- DNS查詢失敗
- 瀏覽器更新後導致的錯誤
- 企業防火牆阻擋、刪除或修改某些內容(常見於銀行、政府部門等大型機構)
- 行動網路供應商為加快載入速度及節省頻寬用量而修改內容
- 個人防火牆或是防毒軟體修改或阻擋某些內容
- 使用不安全的網路連線，網路供應商將自己的網頁程式碼插入到頁面中，導致與原本內容發生衝突

某些使用者刻意關閉瀏覽器的部分功能，您應該尊重他們的決定，並確保這些使用者仍然可以使用您的服務。

相關翻譯

編寫語意標記

<https://html.com/semantic-markup/>

當您學習HTML與Web時，您可能會常常反覆遇到一個未被定義的特定詞彙。這個詞正是「語意(semantic)」。

您可能會讀到像是「我們去尋找語意元素」或「我們嘗試盡可能語意化」之類的說明，但永遠無法清楚地了「語意」一詞的含義。

在本文中，我們將探索語意標記的世界，提出該術語的可行定義，並將概念應用於HTML標記的方式。

什麼是語意標記？

根據Dictionary.com的定義，「語意」是指某個詞語或句子的正確解釋或意義。

從語意上來使用詞語就是依照詞語的真正意義來使用它；當我們錯用詞語時，那麼我們就是在非語意地使用它。

許多HTML標籤具有語意意義。換句話說，這些元素本身就能告訴我們關於其中內容的一些資訊。

例如，當瀏覽器遇到一個h1標籤時，它能解釋此標籤表示h1元素的內容構成了包含該元素段落的主要標題。

h1標籤的語意含義是用於識別特定網頁或段落中最重要的標題。

實現語意標記的兩種實踐方法

如果我們要撰寫語意標記，必須採用兩種不同的實踐方法：

- 語意標記需要依據HTML元素原本預期目的來使用。
- 語意標記需要將內容與呈現分開。

正確地使用HTML元素

在撰寫語意標記時，我們使用HTML標籤來向瀏覽器提供有關元素內容的資訊。在語意標記中，標籤不再僅僅是將內容以人類可讀的格式顯示在網頁上的方式。

這些標籤本身成為了一種向機器(無論是瀏覽器、電腦、智慧型手機或其他智慧型裝置)傳達內容意義的方式。

要撰寫語意標記，我們必須正確使用HTML標籤，以確保我們的標記既可供人類閱讀，又可供機器閱讀。

分離內容與呈現

過去，經常使用標記來定義樣式與網頁布局。

開發者可能會根據網頁在瀏覽器所呈現的視覺效果來選擇標題級別，而不是其真正的階層意義；表格可能被當作頁面布局的工具，而不是用來呈現資料；還有一些HTML標籤，如 `frameset`，其設計初衷就是用於網頁布局。

但當我們撰寫語意標記時，我們不能單純基於視覺呈現來選擇HTML元素，反而應該基於語意來選擇HTML元素，惟後使用CSS來定義內容的視覺呈現。

在撰寫語意標記時，網頁元素的呈現完全獨立於內容本身的標記之外。

定義語意標記

牢記這兩種做法，我們可以這樣定義語意標記：

語意標記是使用標記語言(如HTML)，透過正確選擇標記元素來傳達文件中每個元素的意義，並保持文件中所包含元素的標記與視覺呈現間完全分離。

為什麼語意標記這麼重要？

對一般的網站瀏覽者，良好的CSS可以隱蔽不良的HTML標記。然而，無論多少樣式都不會使不良的標記對於電腦化訪客更有意義，如搜尋引擎的網頁爬蟲、瀏覽器翻譯工具或螢幕報讀軟體等輔助科技。

[Bruce Lawson指出](#)，語意化地使用HTML元素可以「提高其無障礙性、可搜尋性、國際化及互通性(interoperability)」。換句話說，如果您希望您的網站可以對所有人開放、在搜索引擎中獲得較高的排名、服務全球的使用者，並且有效地與其他網站服務進行互動，那麼就必須使用語意標記。

語意標記是為了建立人類與電腦均可閱讀的網頁內容。當網頁能夠被人類與電腦同樣地良好地閱讀時，它就會變得更容易取用，因為電腦便可以更好地分析其內容、索引與交付，而開發者則可以更好地將不同資訊來源整合成新的網頁服務。

如何撰寫具有語意的標記？

我們透過正確選擇與使用HTML標籤，以及挑選能夠傳達標籤所標記內容的資訊標籤來書寫語意標記。

在HTML中有一些是具有語意的元素，而有一些則是非語意元素。非語意元素的例子含 `div`與 `span`。這些標籤並不告訴電腦關於元素內容意義的任何資訊。

雖然非語意標籤在某些情況下很有用，但如果對於特定用途有適合的語意標籤，應優先使用語意標籤，而不要倚賴非語意標籤。

許多語意標籤來自於由Google與Opera等公司對網頁標記的分析。他們觀察到許多網站使用 `id`與 `class` 屬性來提示非語意元素內容含義。

例如，他們發現許多div元素看起來像這樣：<div id="nav">、<div id="header">與<div id="footer">。這些發現可以幫助W3C識別並在HTML5中定義新的語意標籤，如nav、header、footer、article與aside。我們可以將最常見與重要的語意元素分為四類：

- 文件結構標籤
- 文字意義標籤
- 媒體類型標籤
- 關聯性標籤

1. 文件結構標籤

以往div元素是網站中識別與分組各個網頁段落的主要方法。但隨著HTML5的問世，我們有幾個新的標籤可供使用，不僅可以用來分組屬性外，還提供了語意意義：

- header：用於網頁標題的容器元素，通常用於網頁頂部，其中包含了網站標誌、標題元素，以及網站導覽。
- footer：用於網頁頁腳的容器元素，通常用於網頁底部，其中包含了作者資訊、聯絡方式、版權資訊，以及導覽連結與返回網頁頂部的連結。
- main：高層級元素，用於包含單個網頁上唯一且不在多個網頁上重複出現的所有內容。
- nav：用來包含網站導覽連結區塊的元素。通常位於網頁的頂部與底部，也可能在側邊欄(aside)元素中使用。
- section：用來標示文件中的區段，例如章節或長文的主要段落。
- aside：用來識別與網頁主要內容相關但不是文件主要流程的內容。例如，aside元素可以包含部落格文章中某個術語的詞彙定義，或者與頁面內容相關的廣告。
- article：用來識別一個區塊的內容，適合在其他場合進行重複使用與共享，例如部落格文章或技術文章。

請參閱[文件教學](#)，以深入了解如何使用這些為網頁增加結構的語意標籤。

2. 文字意義標籤

在網際網路初期，我們常看到這樣的標記如下：

```
<style> .italics { font-style: italic; } </style>
<p>Some paragraph content including one <span class="italics">itali
cized</span> word.</p>
```

現在，我們通常不會再使用上述的方式來編寫HTML，因為span元素並對瀏覽器或其他電腦訪客來說，無法提供巢狀於開始與結束標籤之間的文字意義或目的的任何資訊。與其使用非語意的span標籤，我們將在應該以斜體顯示的詞語周圍新增em標籤。透過使用em標籤，使用螢幕報讀軟體或存取內容的其他電腦訪客會理解這些標籤是用來強調被標記的內容。em元素只是HTML標籤為文字內容增加語意意義的一個例子。其他例子含：

- h1至h6：標題元素標籤用於識別應該顯示為標題的文字。h1為最高層級或最主要的標題，而h2到h6標題層級則按其重要性排序遞減。

- strong：strong標籤所標記的文字被賦予額外的重要性，通常以粗體字體顯示。
- mark：mark標籤用於強調特定上下文中具有重要性的文字，例如：它可以用於強調搜索結果頁面中所搜索的詞彙。
- cite：cite元素用於識別某段內容的起源。
- blockquote與q：blockquote 與 q(quote)元素用於識別來自其他來源的直接引用文字。
- time：time元素用於告知瀏覽器、網頁爬蟲與其他智慧裝置，特定內容以24小時制或特定的日期呈現。

3. 媒體類型標籤

HTML5也包含了三個標籤，用於識別標籤之間的媒體類型。這些標籤具有兩個目的：首先，它們會通知瀏覽器需要排隊處理使用特定的技術資源，如影片回放引擎；其次，他們為內容指派語意意義。

- audio：用於在文件中增加一個或多個音訊內容來源，並允許瀏覽器根據訪客的設備與瀏覽器選擇最佳選項。
- video：類似於audio元素，但用於將影片內容增加到標記文件中。
- picture：用於允許網頁瀏覽器根據媒體查詢的結果，從可用選項中選擇最佳的影像。

你可以在[HTML5 Media教學中](#)了解更多有關嵌入音訊與影片元素的資訊。此外，我們有一篇關於在網頁上使用圖片的文章，也提供了更多關於何時使用picture元素以及何時保持使用img元素的資訊。

4. 關聯性標籤

在HTML中，有些元素用於顯示多個元素之間的關聯性。例如，使用有序列表(ol)時，即告知瀏覽器該列表內的項目之間彼此有所關聯，應按特定的順序呈現。以下是一些常用於表示多個元素之間關聯性的元素：

- ul：用於表示列表中的項目之間關係，並指示它們無須按特定順序。想要深入了解如何使用有序與無序列表，可以參考[列表教學](#)。
- figure：允許將特定的內容(例詳見圖片、圖表或文本)與由figcaption標籤的標題分組在一起。透過在figure標籤之間巢狀標題與內容，識別了巢狀元素之間的關係。[圖片頁面](#)包含更多實作此標籤資訊。
- address：用於將聯絡資訊與包含address元素的父元素關聯。例如，當新增到文章中時，address元素提供了文章作者的聯絡資訊。當新增到網頁頁尾時，address元素則識別了網頁擁有者的聯絡資訊。

結語

如果您剛開始接觸HTML，建議您投入時間學習如何語意地使用這些不同HTML標籤。您不確定自己是否使用正確的標籤，不妨多花點時間進行研究。正如我們所觀察到的，使用正確的標籤使用是相當重要的。對於已經使用HTML一段時間的人，建議花時間去學習HTML5新加入的元素，以及如何正確地使用它們。過去幾年內，HTML越來越複雜，誘使人們繼續使用帶有class與id屬性的div元素，但語意化的HTML5標籤提供了更好的無障礙性與可交互運作性方面的承諾，足以讓我們接受這些新的語意化元素。隨著網路與智慧型設備的普及，智慧型設備大量湧現，網路漸漸成為現代社會的一部分，讓

我們越發認知到語意標記的重要性。網頁內容不再僅僅局限於桌上型電腦，只能使用少數的瀏覽器存取；如今，語意化網路正在我們周邊迅速成長。確保您所接觸的每一段標記都是語意化，您也為整個網路的持續成長發揮了作用。

相關翻譯

smart answers

<https://insidegovuk.blog.gov.uk/2016/10/27/making-it-easier-to-understand-smart-answer-logic/>

讓答案邏輯更容易理解

內容工具團隊的使命是幫助政府數位服務 (Government Digital Service, GDS) 內的內容設計者能更有效地完成工作，以在多個領域獲得微小的進步，這些微小的進步總與起來會產生巨大的效益。以下部落格文章說明我們為此所做的事情。

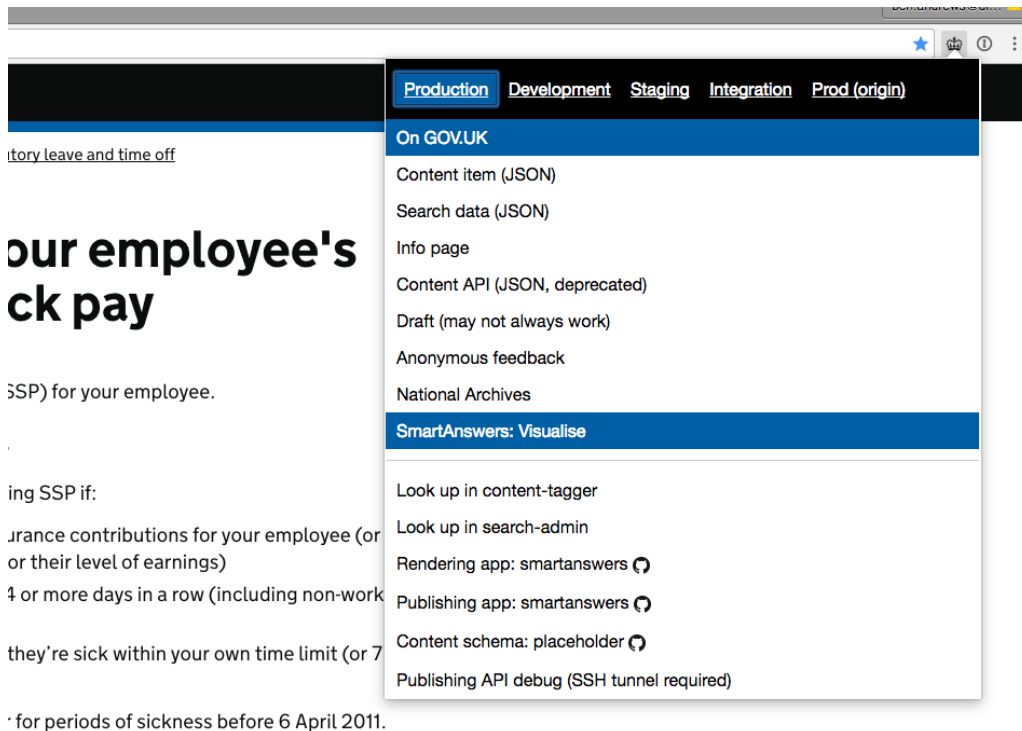
我們在GOV. UK上提供的一個名為「smart answers」的「產品」。smart answers 是我們用來讓複雜政策轉換成為更簡單、更個人化答案的格式。透過轉換成為一系列問題與答案，允許使用者深入了解有時相當複雜的政策，僅提供使用者情況與需求相關的資訊，且不使用專家術語。

更新smart answers 需要部門領域專家、GDS內容設計者與GDS開發人員之間的密切合作。這是為確保內容的準確性、清晰度與smart answers 採用技術限制之間的平衡。這既是內容層面與程式碼邏輯的挑戰，有時需要花費相當多的時間才能做正確。對於外部利害關係人，可能很難理解這些限制並建立一個在幕後運作的「決策邏輯」心智圖。

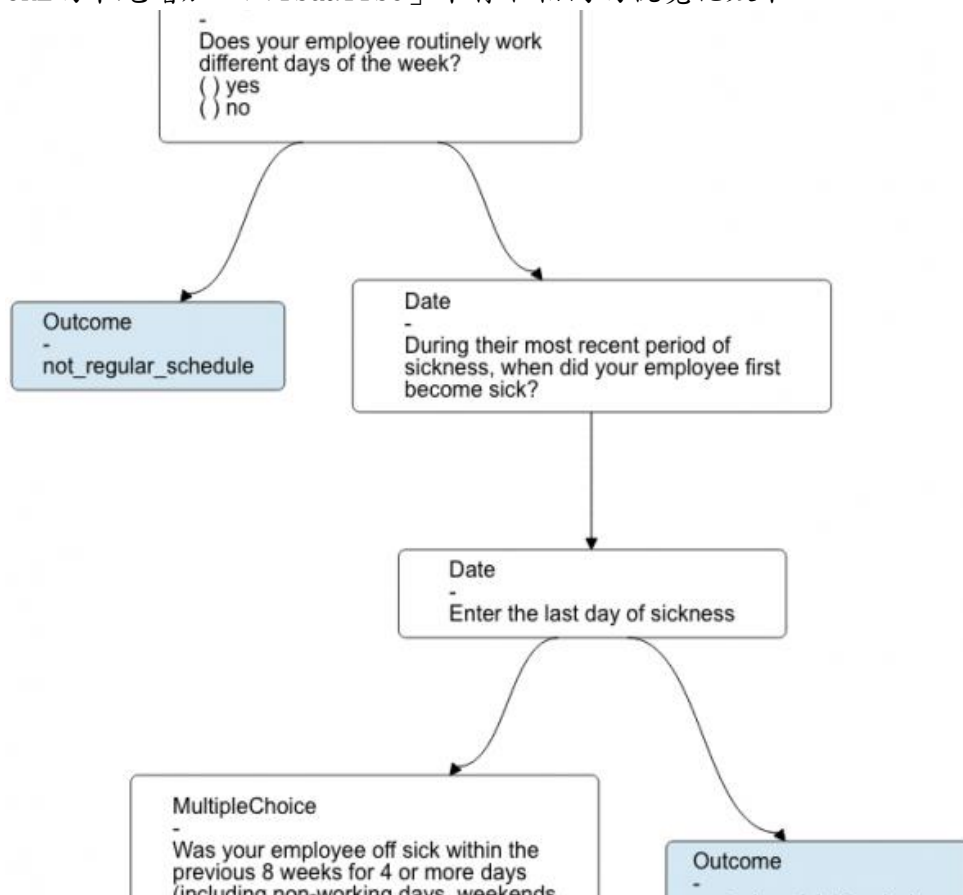
目前三位人員之間的來回過程是透過Google文件進行的，並有一種經過協議的方式來標記與傳達更改。這有一些優點，但缺點是在翻譯過程中可能會失去理解與細節之處。

為了讓事情變得更容易，我們製作一個瀏覽器擴充功能，將邏輯視覺化增加到瀏覽器中的即時smart answers 頁面，並且為了便於使用，我們將其包含在「GOV. UK 工具組擴充功能」。

要查看邏輯視覺化，內容設計者或領域專家所需要做的就是點選瀏覽器擴充功能中的「SmartAnswers: Visualise」，同時點選相關的smart answers 內容。



這會將「[/visualise](#)」新增到任何smart answer URL，惟後在同一瀏覽器分頁中載入邏輯視覺化的解釋。因此，如果無法使用該擴充功能，仍然可以透過在任何smart answer URL的末尾增加「[/visualise](#)」來存取相同的視覺化效果。



這是一個很小的變化，但我們希望有助於逐步降低更新smart answer所需的複雜性與時間。它在大多數情況下效果很好，但對於更大、更複雜的smart answer(例如「在國

[外結婚](#)」網頁)，該頁面可能會有點擁擠。這是我們在下回合的改進中需要考慮的問題。

這只是我們為幫助GOV.UK內容設計者所做工作的範例之一。內容工具團隊還有很多工作要做。

相關翻譯

原始碼(source)的順序與文件的大綱

網頁結構教學

<https://www.w3.org/WAI/tutorials/page-structure/>

良好結構的內容可以實現更有效率的導覽與處理。使用HTML與WAI-ARIA改進網頁與應用程式中的導覽與定向。

- **頁面區域**：使用HTML5與WAI-ARIA識別與標記網頁上的區域。
- **標記區域**：標記區域以允許使用者區分與存取它們。
- **標題**：新增標題並根據標題的關係與重要性對網頁部分進行邏輯巢狀與標記。
- **內容結構**：使用適當且有意義的元素來標記頁面上的內容。

為什麼這很重要？

內容結構良好的頁面對於許多網路使用者來說至關重要，例如：

- **有認知與學習障礙的人**可以更輕鬆地找並優先處理到頁面內容。
- **使用螢幕報讀的使用者**可以直接跳到主要內容並導覽到對他們來說重要的段落。
- **鍵盤使用者**可以更有效地瀏覽頁面及其段落。否則，使用者必須多次按Tab鍵才能瀏覽每個段落中的所有連結。
- **使用僅顯示網頁主要內容的軟體的人**(例如有認知障礙的人)如果正確標記頁面結構，將會獲得更好的結果。
- **有視覺障礙的人**(含視力低下的人)可以在頁面與內容中找到提供定位的提示。
- **行動網路使用者**通常可以使用所謂的「閱讀器」或「閱讀」模式，該模式僅在正確標記的情況下顯示頁面的主要內容。
- **使用某些瀏覽器外掛程式的使用者**可以使用標誌性作用跳到頁面上的特定段落。
- 良好的、無障礙性的頁面結構還有其他好處，超越了身心障礙者的體驗。例如，搜尋引擎可以使用這些資料來更好地索引頁面內容。

相關WCAG資源

這些教學提供了在不同情況下實現輔助功能的最佳實踐指南。本網頁結合了以下WCAG成功標準與來自不同一致性等級的技術：

成功標準：

- **1.3.1 資訊與關係**：透過簡報傳達的訊息、結構與關係可以透過程式設計方式確定或以文字形式提供。(A級)
- **2.4.1 繞過區塊**：提供一個機制，可以繞過多個網頁上重複的內容區塊。(A級)
- **2.4.6 標題與標籤**：標題與標籤描述主題或目的。(AA級)
- **2.4.10 段落標題**：段落標題用於組織內容。(AAA級)

相關翻譯

資料表格

表格教學

<https://www.w3.org/WAI/tutorials/tables/>

資料表格用於以網格形式組織具有邏輯關係的資料。無障礙性的表格需要HTML標記來指示標題儲存格(header cell)與資料儲存格(data cell)並定義它們的關係。輔助科技使用此資訊為使用者提供上下文內容。

標題儲存格必須標記 <th>，資料儲存格必須標記 <td> 以便可以存取表格。對於更複雜的表，可能需要使用scope、id與headers屬性進行明確關聯。

本教學說明如何適當的把結構標記(structural markup)應用於表格。含以下面項：

- [有一個行或列標題的表格](#)：對於易於區分內容的表格，請使用元素標記標題位置，使用<th>元素標記資料格<td>。
- [有兩個標題的表格](#)包含有一個簡單的行標題與一個簡單的列標題：對於標題方向不明確的表格，可以透過將範圍(scope)屬性設為行(col)或列(row)來定義每個標題的方向。
- [有不規則標題的表格](#)具有跨越多個列與/或多行的標題儲存格：對於這些表格，定義列組與行組，並使用範圍屬性的行組 (colgroup) 與列組 (rowgroup) 來設定標題儲存格的範圍。
- [具有多層標題的表格](#)具有每個資料儲存格關聯多個標題儲存格：對於複雜到標題儲存格無法以嚴格水平或垂直方式關聯的表格，請使用id與headers屬性來明確關聯標題儲存格與資料儲存格。
- [標題與摘要](#)：標題標識了表格的整體主題，在大多數情況下都很有用。摘要於複雜的表格中提供方向或導覽提示。

HTML以外的某些文件格式(例如PDF)提供與標記表結構類似的機制。文書處理應用程式還可以提供標記(markup)表格的機制。雖然某些程式可能提供幫助轉換表格標記的功能，但從一種格式轉換為另一種格式時，表格標記經常會遺失。

許多網站設計(web authoring)工具與內容管理系統(CMS)提供在表格建立期間定義標題儲存格的功能，而無需手動編輯程式碼。

註：

本教學提供了建立用於在網格中顯示資料的表格指南。本教學不適用於版面配置的表格。作為一般規則，表格不應該用於版面配置的目的。取而代之的是使用層疊樣式表(CSS)進行視覺呈現才是最佳實踐。

為什麼這很重要？

缺乏結構標記，無法正確連結並區分標題與資料儲存格的表格，會對造成無障礙性阻

礙。僅依靠視覺提示不足以建立無障礙性的表格。透過結構標記，標題與資料儲存格可以由軟體以程式設計方式確定，這意味著：

- **使用螢幕報讀軟體的人**可以在瀏覽表格時大聲朗讀行標題與列標題。螢幕報讀軟體一次讀出一個儲存格並引用關聯的標題儲存格，因此閱讀器不會遺失上下文。
- **有些人使用替代方法來呈現資料**，例如使用自訂樣式表來顯示並強調標題儲存格。此類技術使他們能夠更改文字大小與顏色，並將資訊顯示為列表而不是網格。需要適當地建構表格的程式碼以允許使用替代方法來呈現資料。

相關翻譯

效能問題

為什麼我們要關注前端效能

<https://technology.blog.gov.uk/2019/04/18/why-we-focus-on-frontend-performance/>

英國政府最近更新了[服務手冊指南](#)，以幫助政府各部門的開發人員在效能成為使用者問題之前，識別與測試性能問題。

確保GOV. UK盡可能具有包容性是持續努力的部分。2016年的Google研究發現，如果行動網站載入時間超過3秒，[53% 的使用者就會放棄該網站](#)。對政府而言，GOV. UK通常是使用者可以獲得資訊的唯一地方。如果網站表現不佳，將成為單點故障(single point of failure)。

進一步了解我們的使用者

為了幫助人們盡快從GOV. UK獲得所需的資訊，我們需要了解我們使用者所處的狀況與環境。

不同的設備

不是所有的使用者都負擔得起高階設備，我們已經發現，配備入門級硬體的便宜設備已經難以應對目前的網頁系統。我們定期評估人們使用哪些[設備與瀏覽器來存取GOV. UK](#)，以便確認我們的前端在所有這些裝置與瀏覽器上都能有效運作。

連接速度的差異

即使使用者確實擁有現代化的設備，英國各地的連線速度仍有所不同。

雖然[英國2018年的平均寬頻速度為18.57 Mbps](#)，但網路連線速度可能有很大差異。例如，即使是倫敦這個擁有全球連結與基礎設施的城市，居民網路連線也可能很差。在所謂的「沒有訊號的地方」(not-spots)，居民[連接速度僅有0.26 Mbps](#)。

在這些情況下，還有一些使用者甚至無法依賴行動網路覆蓋率，因為[英國境內有15%的室內空間是沒有行動網路覆蓋](#)。

譯註：英國 15% 的室內空間沒有行動網路的原文2017調查連結已失效，並附上2018英國室內空間沒有行動網路覆蓋已降到8%。

https://www.ofcom.org.uk/_data/assets/pdf_file/0020/130736/Connected-Nations-2018-main-report.pdf

即便使用者有網路，但由於網頁有太多資源，使載入與運行緩慢，因此英國有很多人經常無法存取這些網站。透過壓縮資源，我們可以減少所需的資料傳輸，並為連接速度差與設備較舊的使用者提供更好的服務。

行動網路使用可能很昂貴

此外，Ofcom統計數據顯示，英國[每月行動網路的平均費用為18.36英鎊](#)。該資費每月為使用者提供1.9 GB的上網流量。考慮到[行動網頁量中位數接近1.7 MB](#)，每個月行動上網流量是不夠用的。

透過減少網頁的大小，我們可以改善使用者設備的數據消耗及關聯的電池壽命。

使用者的壓力

最重要的是，糟糕的網站效能會增加使用者的壓力並危害他們的工作效率。Glasgow Caledonian University的一項研究發現，當參與者在連線速度慢的網站內，完成一項簡單任務時，[注意力必須提高50%以上](#)。執行任務所需的額外注意力會導致整體壓力平均上升。

當GOV. UK是使用者可以獲得政府資訊與服務的唯一途徑時，緩慢的網站效能是無法接受的。

我們正在採取哪些措施來提效能

為了對未來有所幫助，除了指導政府開發人員如何提高目前前端的效能外，我們還為所有GOV. UK前端開發尋找更廣泛選擇的技術。

為此，我們正在進行一系列調查，以找出哪些技術對網站效能有重大影響，以便我們可以確定未來工作的優先順序。

我們之前已經討論過[有關字體載入的工作](#)，也在研究以下策略，例如：

- 使用現在所有主流瀏覽器都支援的HTTP/2，以最佳化資源(asset)交付到使用者設備
- 更新字體，以獲得更好的整體壓縮以減少資料使用
- 應用現代壓縮方法(例如[Brotli](#))來減少頁面的大小

隨著調查的進展，我們計畫在部落格中發布我們的調查結果，並將隨著了解更多資訊而更新我們的[前端效能指南](#)。在這個過程中，我們希望政府各地的開發人員能從已開始的工作中學習，並幫助我們確定下一步要研究的內容。

效能是每個人的責任

網站效能可能決定某人是否能夠從政府獲得需要的東西，或完全放棄他們的任務。因此，效能是我們提供的服務不可或缺的一部分，服務團隊的每個成員都應該參與最佳化過程。即使是微小的更動也能為我們的使用者帶來巨大的改變。

相關翻譯

無障礙性

利用輔助科技進行測試

<https://www.gov.uk/service-manual/technology/testing-with-assistive-technologies>

您必須讓服務與輔助科技可以搭配使用，這樣每個人可以透過他們所依賴的技術(例如螢幕報讀器或語音辨識軟體)來使用您的服務。

使用輔助科技進行[整體無障礙測試](#)應該成為您服務的一部分。

可以使用哪些輔助科技進行測試

在整個開發過程中使用輔助科技測試您的服務，尤其是在引入重要功能或進行重大變更時。

符合服務標準

為了滿足服務標準，您的服務在進入公開測試版之前，必須至少與以下輔助科技和瀏覽器組合一起使用。

	軟體版本	瀏覽器
JAWS(桌面螢幕報讀器)	2019年或之後	Chrome 或 Edge(最新版本)
NVDA(桌面螢幕報讀器)	最新版本	Chrome 或 Edge(最新版本)
iOS 上的 VoiceOver (行動螢幕報讀器)	最新版本	Safari(版本 12 或更高版本)
TalkBack (行動螢幕報讀器)	最新版本	Chrome(最新版本)
Windows 放大鏡或 Apple Zoom(螢幕放大鏡)	最新版本	任何
Dragon(語音辨識)	15或以上	Chrome(最新版本)

此表基於 [2016 年於GOV.UK 使用輔助科技的調查](#) 以及最近的 [WebAIM 螢幕報讀器調查](#)。

您可以自行進行此測試，或要求將其作為[無障礙稽核](#)的一部分。

您可能想要測試的其他輔助科技

在可能的情況下，最好使用其他輔助科技、瀏覽器和作業系統設定進行測試。我們建議優先考慮以下項目：

- 舊版的輔助科技——特別是JAWS和TalkBack
- 其他輔助科技——特別是macOS上的VoiceOver(螢幕報讀器)以及 Safari 和 ZoomText(螢幕放大鏡)
- 更改顏色——使用Windows高對比模式和Firefox瀏覽器設置
- 使用瀏覽器和輔助科技的其他組合進行測試

其他需要優先考慮的瀏覽器和輔助科技組合含：

- Firefox(最新版本)和 JAWS
- Firefox(最新版本)和 NVDA
- Firefox(最新版本)和 Dragon

持續測試

在開發過程中越頻繁的測試，就越有可能識別出任何無障礙問題。如果可以的話，最好採用使用者常用的輔助科技持續測試。

但如果這無法落實，[請採用您可以使用的輔助科技進行測試](#)。嘗試包含桌面螢幕報讀器、行動螢幕報讀器、螢幕放大器和一些語音辨識軟體。

如何測試

盡可能使用實際設備或技術進行測試。如果您無法存取設備或技術，請考慮使用虛擬機。

測試時，您應該檢查：

- 您可以存取資訊
- 該資訊是可以理解的
- 介面上的所有內容都可用

螢幕報讀器

您應該使用螢幕報讀器進行以下測試：

- 讀取每個元素和標題
- 逐一瀏覽每個連結
- 檢查每個地標(landmark)，例如頁尾/footer)和任何導覽列(navigation)
- 檢查您對無障礙網際網路應用程式(ARIA)的使用情況
- 檢查您是否可以填寫任何可編輯欄位，例如編寫和提交表單

有關使用螢幕報讀器(含鍵盤命令)的更多資訊，請閱讀以下WebAim文章：

- [使用JAWS評估網站的無障礙性](#)
- [使用NVDA評估網站的無障礙性](#)
- [使用VoiceOver評估網站的無障礙性](#)

螢幕放大鏡

使用螢幕放大鏡時，測試放大倍率至少為4倍。查看：

- 元素之間的間距，例詳見表單標籤和欄位之間的間隙
- 頁面元素在不同的頁面布局上能一致地顯示——例如，放大頁面的人總是可以找到搜尋框
- 使用者可以知道視區(viewport)之外發生的事情——例如，使用互動視窗(modal)或顯示錯誤訊息

語音辨識

若要使用語音辨識技術測試您的服務，請使用語音：

- 導覽至每個功能
- 啟動每個連結、按鈕或互動元素，例詳見表單控制或媒體播放器
- 如果適用於您的服務，請在任何表格中輸入文字

確保您說話清晰、自然。您還應該使用高品質的耳機麥克風，而不是電腦的內建麥克風，並確定您與麥克風的距離保持一致。大聲說出標點符號並更正任何口誤。

請查看 [Dragon的使用者指南](#)，以了解有關如何安裝、發出語音命令和聽寫不同類型文字的更多資訊。

如果發現問題該怎麼辦

如果您在測試時發現問題，請清楚地記錄下來，以便可以複製和重新測試。

在記錄問題或花時間嘗試解決問題之前，請確保：

- 已正確使用輔助科技
- [程式碼沒有任何漏洞或錯誤\(bugs or errors\)](#)

- 瀏覽器或輔助科技中沒有已知可能會影響測試的問題

記錄問題時，您應該描述：

- 問題
- 問題影響的對象
- 測試時使用的瀏覽器、作業系統和技術版本
- 附上螢幕截圖可能會很有幫助。

如果您正在記錄多個問題，將它們按嚴重程度分類以便進行優先處理可能有所幫助。例如，如果您發現一個問題阻礙了某一組使用者的使用，您可能會希望將此問題列為比使用者可以輕鬆解決的問題更高的優先順序。

考慮：

- 與更廣泛的[無障礙社群分享您的發現](#)以便其他人可以從您的測試中學習
- 向瀏覽器或輔助科技供應商報告任何錯誤

將輔助科技使用者納入您的使用者研究中

您還應該將輔助科技的使用者納入您的使用者研究中。參與者不需要使用任何特定的軟體組合——任何設置都有助於了解您的服務與輔助科技一起運作的情況。

相關翻譯

實際無障礙的使用者

與身心障礙人士一起進行研究會議

<https://www.gov.uk/service-manual/user-research/running-research-sessions-with-people-with-disabilities>

當您對身心障礙者進行研究時，仔細規劃您的會議、尊重參與者並考慮他們的個人需求非常重要。

遵循的步驟

在會議開始之前與進行中，您需要思考採取哪些措施才能使參與者時感到輕鬆舒適。

會議前

如果您在研究實驗室或其他場所辦理會議：

- 在接待處與參與者會面(或安排其他人)
- 帶他們到實驗室或研究室——了解[如何引導盲人或弱視人士](#)
- 詢問他們是否需要您設置他們帶來的任何設備——這可能需要時間，所以請確定您已做好計畫
- 說明最近的廁所在哪裡，含無障礙廁所
- 說明如果火警響起會發生什麼狀況

如果您的參與者需要使用輔助科技，請要求他們隨身攜帶。因為很難在不同的裝置上重新建立個人設定。如果參與者無法攜帶他們的輔助科技，最好親自拜訪他們。

不要撫摸參與者的導盲犬、輔助犬，除非他們說您可以這樣做。

在開始會議之前：

- 自我介紹並解釋與您在一起的其他人(例如觀察員或記錄員)以及他們的角色
- 確保他們可以使用並理解同意書，以便您[獲得他們的知情同意](#)

會議期間

無論您正在什麼研究，使用哪種[使用者研究方法](#)：

- 直接與參與者交談，而不是與他們的口譯員或助手交談
- 說話清晰並使用日常用語，無需擔心引起冒犯——例如：盲人或弱視人士會使用「下次見」與「明白我的意思」等常用片語
- 確認他們是否需要您大聲說話或放慢速度
- 如果參與者講不清楚，請勿猜測他們在說什麼——如果需要，請他們寫下來
- 不要猜測或假設他們能做什麼或不能做什麼——在幫助他們做某事之前先詢問
- 不要猜測或假設他們的感受或他們為什麼做某事——如果您想知道，請詢問
- 如果您不確定如何提及參與者的殘疾或損傷，請詢問他們喜歡什麼術語
- 如果不清楚，請詢問他們的狀況如何影響他們對科技的使用

會議結束後

研究會議結束後，感謝參與者的參與。

如果您在研究實驗室或其他場所：

- 確定參與者收拾好他們帶來的所有輔助科技
- 確認他們是否需要您為他們預定出租車或安排專人接他們
- 將他們帶回接待處(或請其他人協助)

範例與案例研究

閱讀有關[如何針對視障使用者測試護照服務](#)的部落格文章。

在《問就對了：在整個設計中融入無障礙》(Just Ask: Integrating accessibility throughout design) 一書中[了解有關與身心障礙人士互動的更多資訊](#)。

海報

參考由英國內政部製定的研究訪問需求：「[何時含誰](#)」(who to include when)

相關翻譯

前端的效能

如何測試前端效能

<https://www.gov.uk/service-manual/technology/how-to-test-frontend-performance>

效能不佳的網站可能會讓使用者感到沮喪，並耗盡他們的行動網路數據與電池壽命。您可以透過確保您的前端服務，盡可能快速且輕量來解決這個問題。

這也意味著您的服務可以在多種裝置上正常運作。

為什麼前端效能很重要

透過專注於前端效能，您可以提高服務的速度與整體使用者體驗。這將允許使用者快速有效地完成他們需要做的事情，並且您將擁有易於維護的高品質程式碼。

每個人都可以使用政府服務非常重要。提高前端效能意味著您盡可能的包容以下使用者：

- 無法負擔最新設備的使用者——因為即使在較舊、速度較慢的硬體上，您的服務也會載入得更快
- 使用有限行動網路數據的使用者——因為您的服務將使用更少的行動數據
- 居住在行動網路訊號較差地區的使用者

具有最佳化前端效能的服務也更適合環境，因為它們消耗的電量更少。由於使用者不必經常升級設備來滿足性能需求，因此被棄用的設備會更少。

使用性能作為指標

最佳化前端效能不僅僅是開發人員的任務。整個服務團隊都應該要：

- 將前端效能納入在服務文件中
- 盡早在瀏覽器中進行原型設計(prototyping)與設計——這有助於早期發現性能與設計問題。
- 在各種設備上進行測試——這可以幫助您發現僅在高階筆記型電腦上測試時找不到的問題
- 建立原型以展示設計決策的影響，並突顯效能問題

設定績效預算

績效預算是您為網頁特定功能設定限制的地方，可以是特定的載入時間、網頁發出的請求數量、頁面的大小或這些因素的組合。

一旦設定了預算，您就需要遵循它。每次修改頁面或新增內容時，請檢查是否仍在預算之內。如果超出預算，您可以透過以下方式解決此問題：

- 最佳化頁面上的功能
- 從頁面中刪除功能
- 不將新功能增加到網頁中

績效預算提供了整個服務團隊可以使用的架構。這將有助於引導團隊圍繞在網頁可以新增哪些功能的對談，還能使團隊能夠清楚地解釋他們對服務前端(frontend)所做的決策。

在設計服務過程中，儘早設定績效預算非常重要。在建立服務的過程中途引入績效預算會限制其效果，因為團隊可能必須回去修訂之前設計頁面的績效預算。

如何運行效能測試

您可以使用免費的測試工具來測試服務的前端效能：

- [Google lighthouse](#)[Chrome DevTools](#)：分析運行時效能
- [Chrome DevTools](#)：檢查網路活動
- [Webpage Test](#)
- [PageSpeed Insights](#)
- [Sitespeed.io](#)

您應該在整個開發過程中衡量效能。設定效能指標，並將效能測試整合到發布過程中，可以更輕鬆地處理出現的效能問題。

開發人員可以遵循的簡單效能工作流程如下：

1. 使用一種或多種工具對目前效能進行基礎測試。
2. 最佳化您發現的任何瓶頸。
3. 量化出最佳化前後的效能差異與使用者體驗。
4. 重複此操作，直到您的前端達到所需的效能指標。