



**COLEGIO DE CIENCIA Y TECNOLOGÍA**

LICENCIATURA EN INGENIERÍA DE SOFTWARE

**APÉNDICE 2: DESARROLLO DINÁMICO CMS DNN  
PORTAL WEB DEL COLEGIO DE CIENCIA Y TECNOLOGÍA**

TRABAJO RECEPCIONAL PARA OBTENER EL GRADO DE:  
**LICENCIADO EN INGENIERÍA DE SOFTWARE**

PRESENTA:

**JESÚS ARTURO CISNEROS CANTERO**

**CARLOS ALFREDO HARO SARIO**

DIRECTOR DE TESIS:

**M. EN C. ARMANDO VALERA PAULINO**

## Tabla de Contenido

1. Uso del CMS DNN .....	3
2. Requisitos de instalación.....	5
2.1 Crear una base de datos SQL Server para el portal web .....	6
2.2 Seguridad de usuario SQL.....	6
3. Instalación IIS (Internet Information Services).....	16
3.1 Configuración del servidor .....	17
4. Instalación de DNN (CMS) .....	18
5. ¿Qué es módulo? .....	19
6. Creación de Tema.....	22
6.1 Tipos de temas .....	23
6.2 Contenedor .....	24
6.3 Crear un archivo CSS .....	25
6.4 Componentes de un tema.....	26
6.5 Creación del Skin para el CMS DNN .....	27
7. Diseños y Contenedores .....	29
7.1 Contenedor y diseño alternativo .....	30
7.2 Diseño de portal y contenedor .....	30
7.3 Diseño de página y contenedor .....	31
7.3.1 Contenedor de módulo.....	31
7.3.2 Contenedor de panel.....	31
7.4 Hojas de estilo en cascada (CSS) .....	31
7.5 Crear un XML de tipo de documento .....	33
7.6 Empacar la extensión .....	34
7.6.1 Pasos .....	34
8. Administración del CMS DNN .....	37
8.1 Configuración del DNN .....	37
9. Seguridad .....	48

9.1 El JSON Web Token (JWT) ..... 48

## 1. Uso del CMS DNN

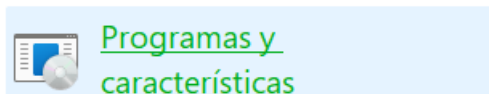
Algunos de nuestros lectores pueden estar familiarizados con el uso del algún CMS, sin conocer otras alternativas de gestores de contenido, como se descubrió en el apartado de comparación de CMS, si bien se pudo haber quedado en esta primera etapa y posteriormente otra persona con conocimiento técnico, podría continuar actualizando el portal web.

La idea es que la persona con o sin conocimiento técnico, pueda ir nutriendo la información conforme a las necesidades de la UACM.

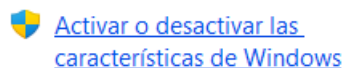
En este punto se debe tener instalado en nuestro panel o servidor web y PC, esta última con el fin de personalizar y crear nuestros aplicativos para esta plataforma. Previamente se describe los pasos en **¡Error! No se encuentra el origen de la referencia.** e 3. Instalación IIS (Internet Information Services)

Se debe activar el servicio de Internet Information Services en nuestro sistema operativo Windows.

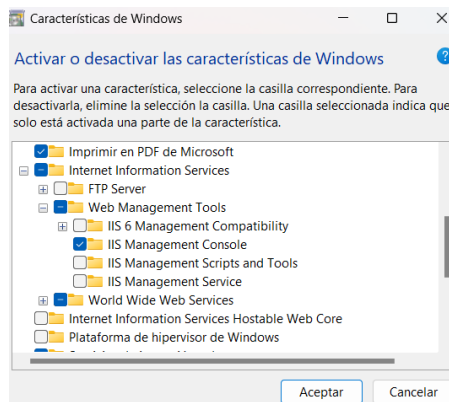
- En nuestro panel de control ir a programas y características



- Activar o desactivar características



- Activar la característica de Internet Information Services

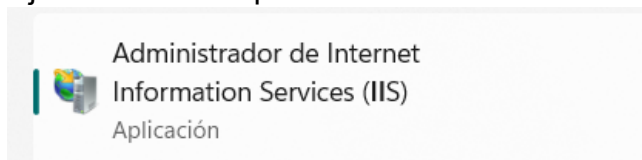


*Ilustración 12. Activar características*

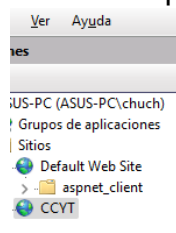
Damos clic en aceptar y reiniciamos nuestro equipo o servidor.

### 3.1 Configuración del servidor

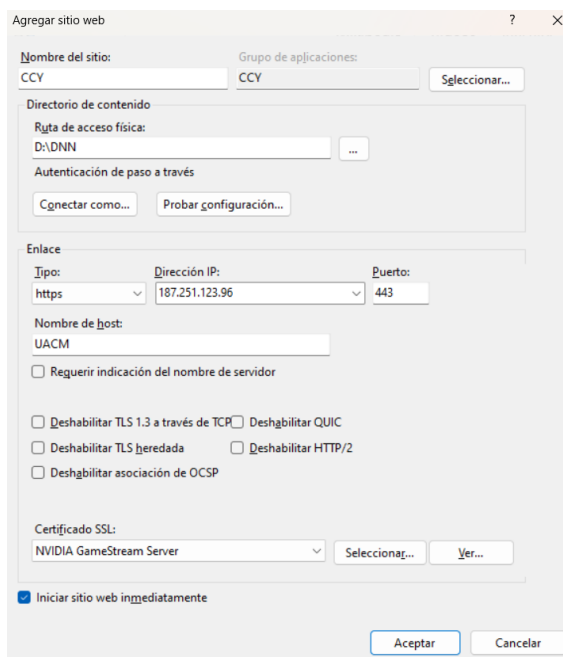
- Ejecutamos la aplicación: Administrador de Internet Information Services.



- Creamos el portal web.



- Configuramos el nombre del portal web, dirección IP, puerto y certificado de seguridad.



*Ilustración 13. Agregar sitio web*

4. Instalación de DNN (CMS). Al instalarse por primera vez se puede elegir un tema en blanco o elegir uno que viene de forma predeterminada.

## 2. Requisitos de instalación

El sistema gestor de contenido DNN necesita que tengamos instalado el sistema operativo del mismo fabricante, en específico:

- “Windows 8, 10 y 11
- Servidor Windows 2012 R2, 2016, 2019 y 2022

Net Framework

- 4.7.2 o superior

Servidor Web

- Microsoft IIS 7.5, 8, 8.5 y 10

Servidor de base de datos

- Microsoft SQL Server 2008 R2
- Microsoft SQL Server 2008 Express R2
- Microsoft SQL Server 2012
- Microsoft SQL Server 2012 Express
- Microsoft SQL Server 2014
- Microsoft SQL Server 2014 Express
- Microsoft SQL Server 2016 (para DNN 8.0.4+)
- Microsoft SQL Server 2017
- Microsoft SQL Server 2019
- Base de datos SQL de Microsoft Azure

Navegador

- Google Chrome
- Mozilla Firefox
- Microsoft Edge
- Entre otros

IDE.

Es necesario este editor para creación extensiones

Visual Studio 2015, 2017, 2019, 2022 (*wrET, s/f*).” (DNN, 2022a)

## 2.1 Crear una base de datos SQL Server para el portal web

En la parte inferior izquierda tenemos un panel de navegación, exploramos los objetos, damos clic derecho en bases de datos y seleccionamos nueva base de datos.

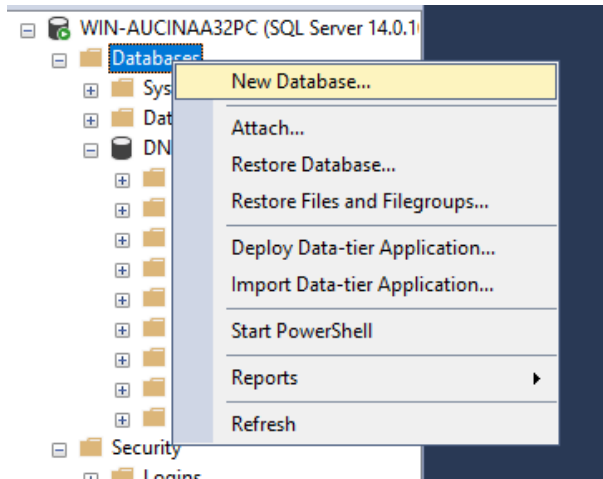


Ilustración 1. Creación de base de datos

Introducimos el nombre de la base de datos y haga clic en Aceptar.

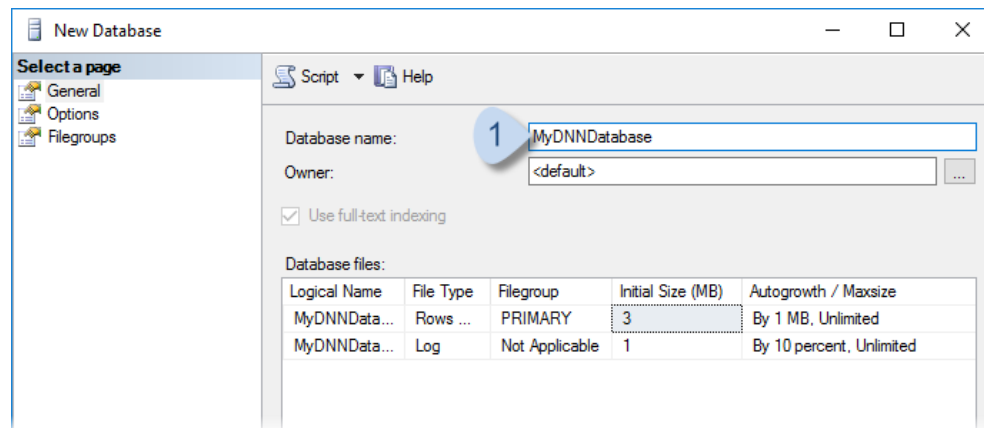


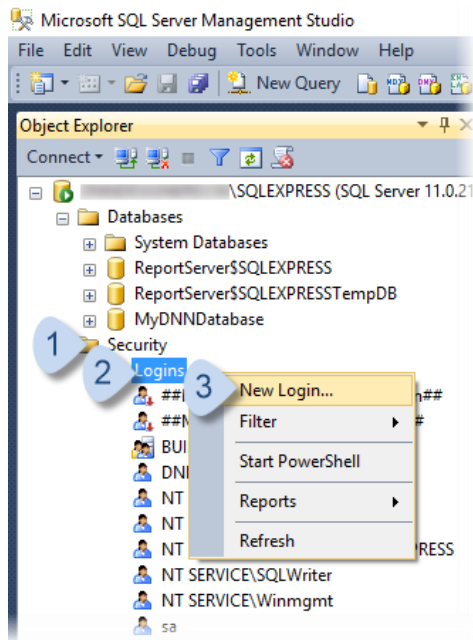
Ilustración 2. Confirmación de creación de base de datos (DNN, 2022b)

## 2.2 Seguridad de usuario SQL

En el panel de objetos:

1. Expandamos las opciones de seguridad.
2. Vamos a la opción de "Logins".

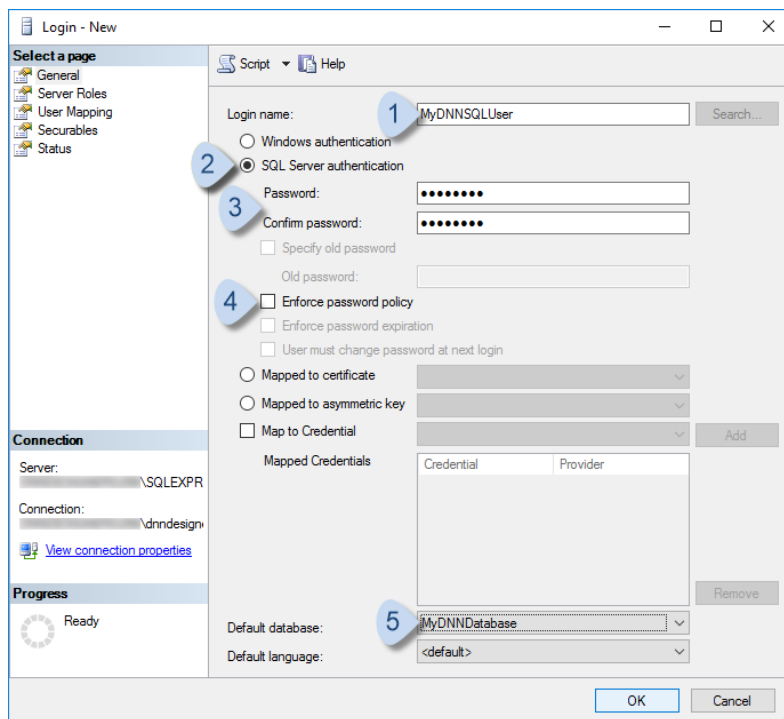
3. Seleccionamos clic derecho y en las opciones damos clic en “New Login”



*Ilustración 3. Nuevos inicios de sesión*

Ingresamos un nombre de usuario.

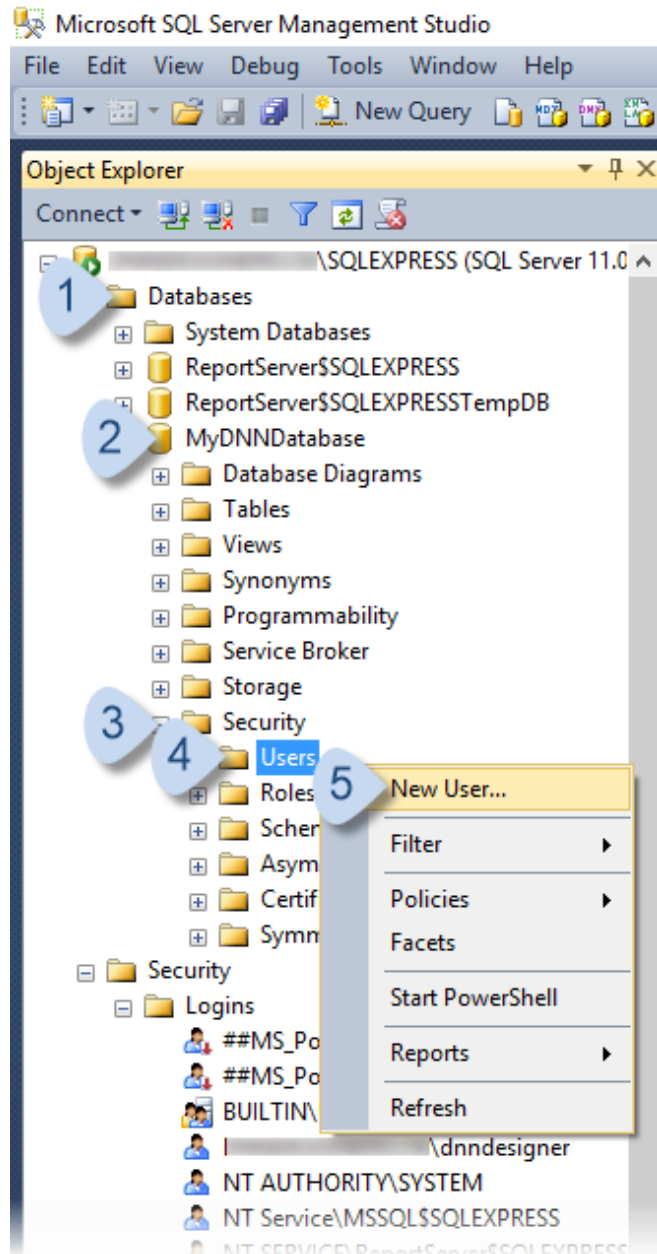
1. Seleccionamos SQL autenticación.
2. Ingresamos una contraseña y la confirmamos.
3. Desmarcamos la casilla de hacer cumplir la política de contraseñas.
4. Seleccionar la base de datos que vamos a usar para nuestro CMS.



*Ilustración 4. SQL autenticación (DNN, 2022b)*

En el panel inferior izquierdo, explorador de objetos.

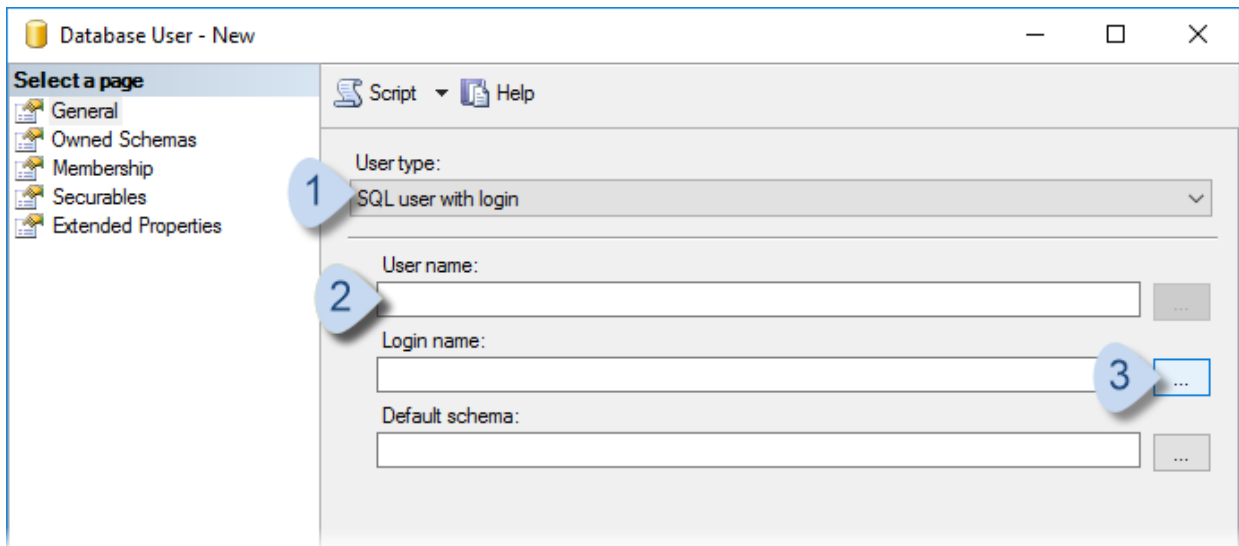
1. Seleccionamos nuestra base de datos creada.
2. Expandimos las opciones de nuestra base de datos.
3. Expandimos las opciones de seguridad.
4. En la carpeta de usuarios damos clic derecho.
5. Creamos un nuevo usuario.



*Ilustración 5. Nuevo usuario (DNN, 2022b)*

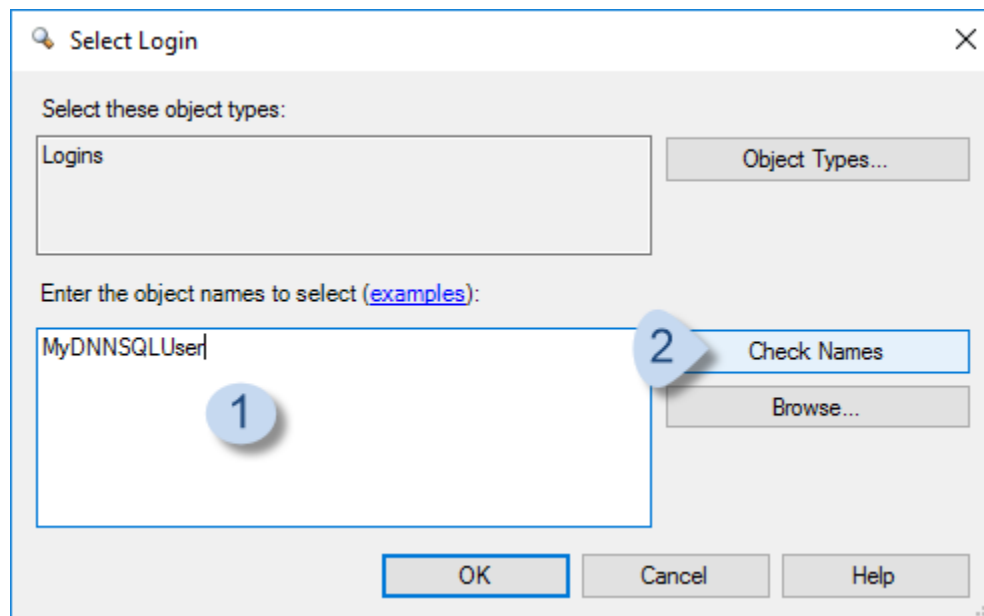
Ingresar credenciales de inicio de sesión:

1. Establezca el tipo de usuario.
2. Nombre de usuario.
3. Clic para verificar el nombre de usuario.



*Ilustración 6. Usuario base de datos (DNN, 2022b).*

En el paso (3) al dar clic nos aparece la siguiente ventana, e ingresamos el nombre de usuario de la cuenta que acaba de crear, luego haga clic o toque Check Names.



*Ilustración 7. Checar nombre de usuario (DNN, 2022b).*

En el apartado “Select a page” seleccione Membership y verifique la membresía del rol de la base de datos que este seleccionado db\_owner (Ilustración 8. Asignar permisos (DNN, 2022b)).

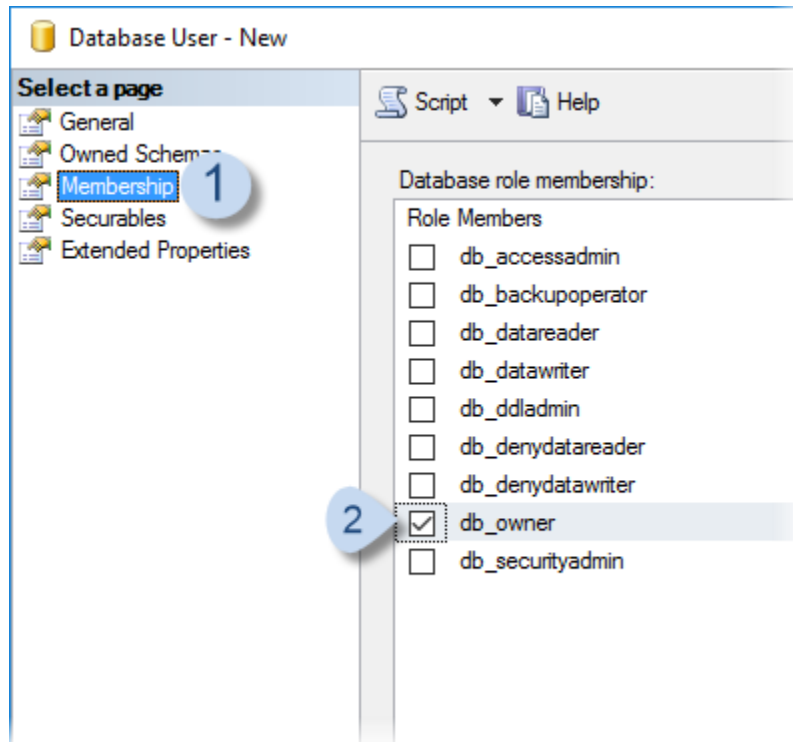
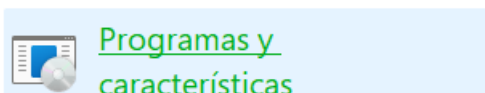


Ilustración 8. Asignar permisos (DNN, 2022b)

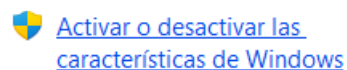
Una vez realizada la configuración de nuestra base de datos tendremos uno de los requerimientos para la 3. Instalación IIS (Internet Information Services)

Se debe activar el servicio de Internet Information Services en nuestro sistema operativo Windows.

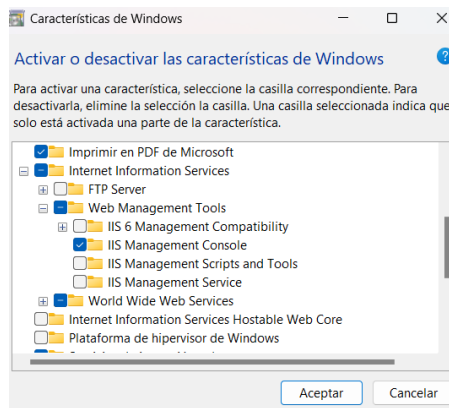
- En nuestro panel de control ir a programas y características



- Activar o desactivar características



- Activar la característica de Internet Information Services

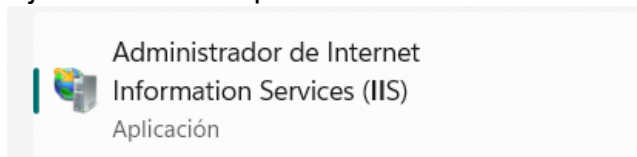


*Ilustración 12. Activar características*

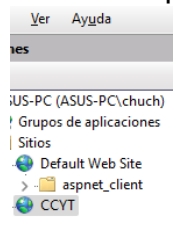
Damos clic en aceptar y reiniciamos nuestro equipo o servidor.

### 3.1 Configuración del servidor

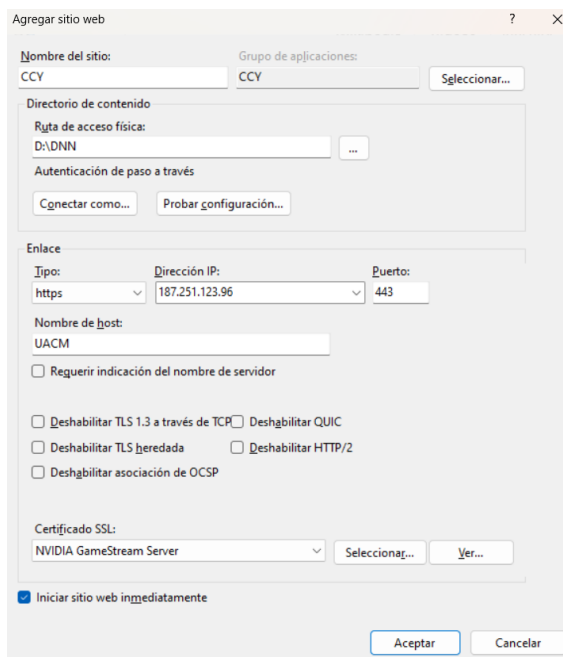
- Ejecutamos la aplicación: Administrador de Internet Information Services.



- Creamos el portal web.

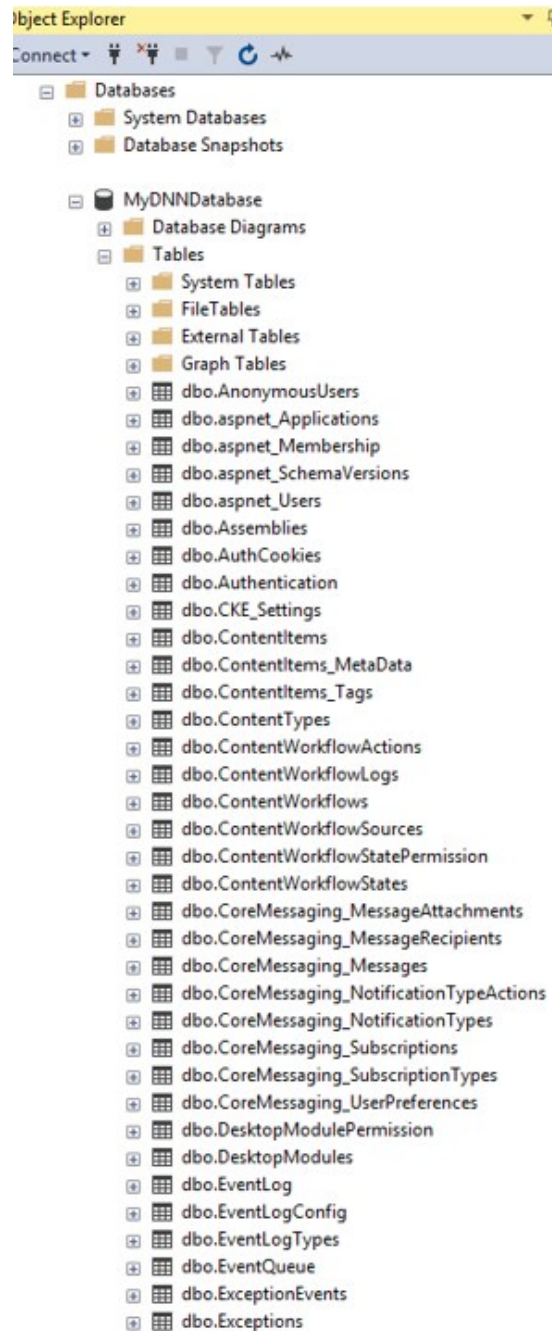


- Configuramos el nombre del portal web, dirección IP, puerto y certificado de seguridad.



*Ilustración 13. Agregar sitio web*

4. Instalación de DNN (CMS), el cual generará una serie de tablas de manera automática donde la creación de los objetos y la información de esta son administrados por el sistema del DNN. En las ilustraciones 9,10 y 11 se muestran las tablas generadas después de la instalación del DNN.



*Ilustración 9. Tablas generadas por DNN parte 1*

+	db	dbo.PersonaBarExtensions
+	db	dbo.PersonaBarMenu
+	db	dbo.PersonaBarMenuDefaultPermissions
+	db	dbo.PersonaBarMenuPermission
+	db	dbo.PersonaBarPermission
+	db	dbo.PortalAlias
+	db	dbo.PortalDesktopModules
+	db	dbo.PortalGroups
+	db	dbo.PortalLanguages
+	db	dbo.PortalLocalization
+	db	dbo.Portals
+	db	dbo.PortalSettings
+	db	dbo.Profile
+	db	dbo.ProfilePropertyDefinition
+	db	dbo.RedirectMessages
+	db	dbo.Relationships
+	db	dbo.RelationshipTypes
+	db	dbo.RoleGroups
+	db	dbo.Roles
+	db	dbo.RoleSettings
+	db	dbo.Schedule
+	db	dbo.ScheduleHistory
+	db	dbo.ScheduleItemSettings
+	db	dbo.SearchCommonWords
+	db	dbo.SearchDeletedItems
+	db	dbo.SearchIndexer
+	db	dbo.SearchStopWords
+	db	dbo.SearchTypes
+	db	dbo.SiteLog
+	db	dbo.SkinControls
+	db	dbo.SkinPackages
+	db	dbo.Skins
+	db	dbo.SQLQueries
+	db	dbo.SynonymsGroups
+	db	dbo.SystemMessages
+	db	dbo.TabAliasSkins
+	db	dbo.TabModules
+	db	dbo.TabModuleSettings
+	db	dbo.TabPermission
+	db	dbo.Tabs
+	db	dbo.TabSettings
+	db	dbo.TabUrls
+	db	dbo.TabVersionDetails
+	db	dbo.TabVersions
+	db	dbo.Taxonomy_ScopeTypes
+	db	dbo.Taxonomy_Terms

*Ilustración 10. Tablas generadas por DNN parte 2*

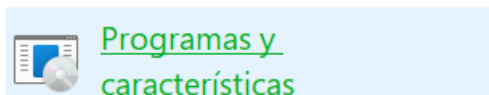
+	+	dbo.Taxonomy_Vocabularies
+	+	dbo.Taxonomy_VocabularyTypes
+	+	dbo.UrlLog
+	+	dbo.UrIs
+	+	dbo.UrlTracking
+	+	dbo.UserAuthentication
+	+	dbo.UserPortals
+	+	dbo.UserProfile
+	+	dbo.UserRelationshipPreferences
+	+	dbo.UserRelationships
+	+	dbo.UserRoles
+	+	dbo.Users
+	+	dbo.UsersOnline
+	+	dbo.Version
+	+	dbo.WebServers
+	+	dbo.Workflow
+	+	dbo.WorkflowStates

*Ilustración 11. Tablas generadas por DNN parte 3*

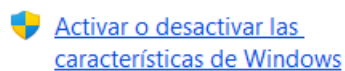
### 3. Instalación IIS (Internet Information Services)

Se debe activar el servicio de Internet Information Services en nuestro sistema operativo Windows.

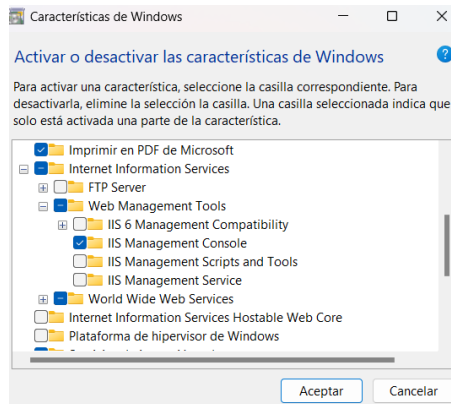
- En nuestro panel de control ir a programas y características



- Activar o desactivar características



- Activar la característica de Internet Information Services

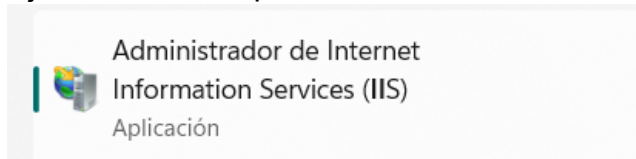


*Ilustración 12. Activar características*

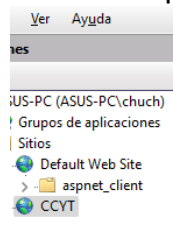
Damos clic en aceptar y reiniciamos nuestro equipo o servidor.

### 3.1 Configuración del servidor

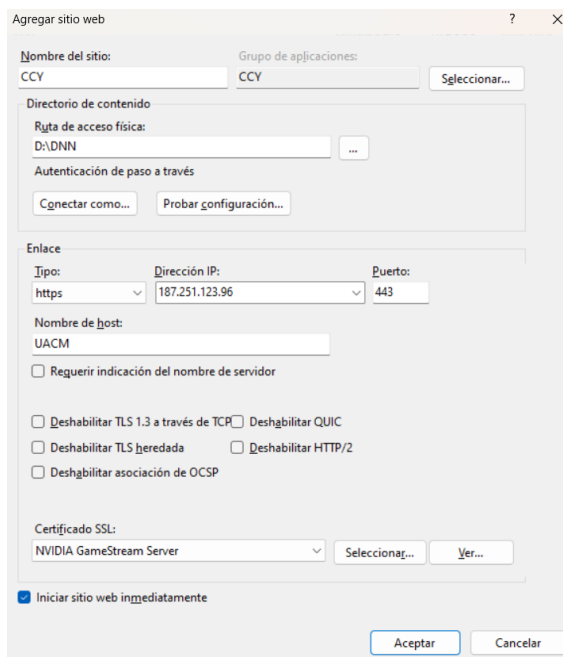
- Ejecutamos la aplicación: Administrador de Internet Information Services.



- Creamos el portal web.



- Configuramos el nombre del portal web, dirección IP, puerto y certificado de seguridad.



*Ilustración 13. Agregar sitio web*

#### 4. Instalación de DNN (CMS)

Una vez que tengamos instalada la base de datos, servidor web, el IDE y el navegador web en nuestra computadora o servidor procedemos a la instalación del gestor del contenido, a continuación, se describen los pasos de la instalación.

1. Ingresar al sitio web del desarrollador <https://www.dnnsoftware.com/community/download>

De las diferentes opciones escogemos DNN\_Platform\_9.10.2\_Install.zip o una versión más reciente si existe.

La siguiente ilustración muestra las diferentes opciones de descarga:

**INSTALL PACKAGE**

**Just the files needed for a run-time deployment**

Download the Install Package if you want to deploy a live site to a web server with the minimum required files.

[DOWNLOAD](#) File Hash: 65c313fc0aefeb3bf597e78283897b9c

**SOURCE CODE**

**Full application source code.**

Download if you are a web developer interested in the DNN Platform web application framework source code.

[DOWNLOAD](#) File Hash: bcf9d83c5e0b97e7681460cca667552

**UPGRADE PACKAGE**

**Files needed for upgrading an existing DNN installation.**

Download this package to upgrade an existing installation to a newer version. This should not be used if you are upgrading an Evoq, DNN Pro or DNN Enterprise installation

[DOWNLOAD](#) File Hash: 5a5bb3bc82cb3f1ffdea2336129561e7

**SYMBOLS PACKAGE**

**PDB and XML files that aid in DNN module development and in DNN debugging.**

Download if you are a developer in need of platform symbols.

[DOWNLOAD](#) File Hash: 6f93c5e32f9aea0d1ababb75b8724931

*Ilustración 14. Página de descarga*

## 5. ¿Qué es módulo?

“Un módulo DNN® es un componente desarrollado con la tecnología net con interfaz de usuario que procesa solicitudes y genera un contenido dinámico” (TODODNN., 2022). Existe muchos módulos, algunos son gratuitos, otros de pago e inclusive podemos desarrollar nuestros propios módulos que se ajusten a nuestras necesidades. A continuación, se mencionan los módulos que usamos en nuestra página web:

“Banners”.

Son conjunto de imágenes ordenadas de forma aleatoria, las imágenes pueden ser de varios tamaños dependiendo del tipo de banner.

“Documents”.

Este módulo consiste en presentar una lista de archivos para que el usuario final pueda descargar, estos pueden estar en nuestro servidor propio o sitios externos, el acceso puede gestionarse con los roles de seguridad del DNN, cuenta con las funciones adicionales:

- Selección de columnas a visualizar.
- Ordenación de documentos por fecha de creación.
- Iconos en documentos.
- Posibilidad de seleccionar una lista por categoría de documento.

“Text/HTML.”

El módulo Text/HTML es utilizado para diversos fines desde mensajes de bienvenida hasta imágenes con breves descripciones, gracias al potente Editor el cual dispone de todas las barras de herramientas para dar formato. En caso de saber lenguaje de etiquetas HTML, dando clic en la parte inferior donde dice source, podemos introducir directamente el código.

1. La carpeta comprimida puede subirse a nuestro servidor net o moverse a nuestro servidor local.
2. Una vez que nuestra carpeta este en nuestro servidor web procedemos a descomprimir los archivos.
3. Iniciamos el servidor.
4. Ingresamos por medio del navegador a nuestro servidor web.
5. Ingresaremos al siguiente asistente.
6. Al iniciar el proceso de instalación la ruta debe ser algo parecida a la que se indica en la parte de abajo, se sustituye por nuestra IP propia o en su caso nuestro dominio. <https://uacm.edu.mx/ccyt/Install/InstallWizard.aspx>

**Administrative Information**

Username \* ⓘ  ✓

Password \* ⓘ  ✓

Confirm \* ⓘ  ✓

Email address: \* ⓘ  ✓

**Website Information**

Website Name \* ⓘ  ✓

Template ⓘ  ▼

Language ⓘ  ▼

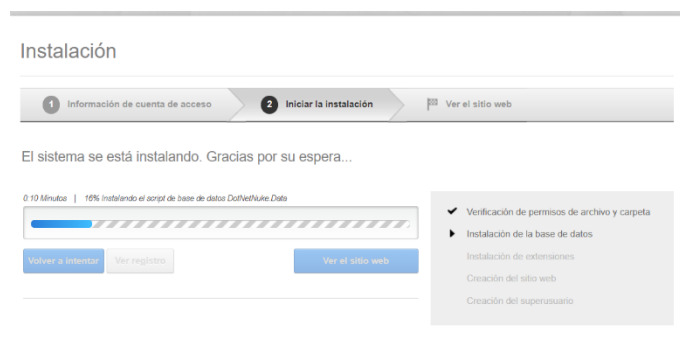
**Database Information**

Database Setup ⓘ  Default  Custom

[Continue](#)

*Ilustración 15. Administrar información*

Procedemos a ingresar los datos del formulario, es importante que se cuente con un manejador de base de datos, por default seleccionará los valores predeterminados o podremos seleccionar “Custom” y pondremos los parámetros tales como nombre de la base de datos y contraseña (Ilustración 15. Administrar información).



*Ilustración 16. Instalación del DNN*

En seguida el sistema se estará instalando (Ilustración 16. Instalación del DNN) y una vez terminada la instalación podremos revisar los logs en caso de que existiera un error en la instalación, si todo salió de manera correcta podemos ir al sitio web, con el sistema web ya instalado.

Para acceder a la administración del sitio accedemos con el nombre del sitio/login. (Ilustración 17. Iniciar sesión)

<https://uacm.edu.mx/ccyt/login>

Nombre de usuario:

Contraseña:

Recordar inicio de sesión

*Ilustración 17. Iniciar sesión*

## 6. Creación de Tema

Si bien no existe la manera de exportar el trabajo realizado en la primera etapa, se puede migrar parte de la estructura en un tema.

A un tema se le llamaba Skin actualmente desde la versión 8 del DNN recibe el nombre de tema, el cual indicará los diversos diseños de las diferentes páginas que se ocupan para la creación del portal web del Colegio de Ciencia y Tecnología, el cual tiene contenedores para incrustar módulos.

Los encargados de incrustar los módulos será el personal con acceso a la administración del portal web del Colegio de Ciencia y Tecnología, que corresponde al encargado del departamento de Informática, dependerá de la cantidad de módulos que tenga instalado en el sistema del DNN, así como de los paneles que disponga la plantilla.

Un tema puede contener diferentes plantillas, en cual permite tener mayor variedad de opciones, una plantilla de diseño define en donde se ubicarán los paneles en el portal web, las plantillas de diseño web pueden ser elegidas para cada página web como en nuestro caso la página principal contiene un diseño diferente a las que se le va a poner contenido.

Una plantilla puede tener uno o más paneles, un contenedor tiene exactamente un panel; los paneles adicionales se ignoran por el CMS.

Se debe de llamar a unos de los paneles ContentPanel su nombre es sensible a mayúsculas y minúsculas (case sensitive), cada panel debe tener un nombre que sea único dentro de la página.

Todos los paneles deben definirse como un control de servidor agregando la palabra a nuestro elemento HTML mediante la siguiente palabra reservada `runat="server"`.

Un panel puede ser agregado dentro del siguiente elemento HTML.

Dentro de una columna en la etiqueta `<td>`

Dentro de un contenedor `<div>`

Dentro de un párrafo `<p>` y

Dentro de una sección de texto con estilo `<span>`

Ejemplo:

## HTML

```
<div id="wrapper">
  <div id="header">
    [LOGO]
  </div>
  <div id="sidebar">
    <div id="Feed">
      <a class="feed-button" href="#"> </a>
    </div>
    <div id="SidebarPane" class="SidebarPane" runat="server"></div>
    <div id="sidebar-bottom"> </div>
  </div>
  <div id="content">
    <div id="BannerPane" class="BannerPane" runat="server"></div>
    <div id="ContentPane" runat="server"></div>
  </div>
  <div id="footer"></div>
</div>
```

## ASCX

```
<%@ Control language="vb" AutoEventWireup="false" Explicit="True" Inherits="DotNetNuke.UI.Skins.Skin" %>
<%@ Register TagPrefix="dnn" TagName="LOGO" Src="~/Admin/Skins/Logo.ascx" %>
  <div id="wrapper">
    <div id="header">
      <dnn: LOGO runat="server" id="dnnLOGO" />
    </div>
    <div id="sidebar">
      <div id="feed">
        <a class="feed-button" href="#"> </a>
      </div>
      <div id="MySidebarPane" class="SidebarPane" runat="server"></div>
      <div id="sidebar-bottom"> </div>
    </div>
    <div id="content">
      <div id="MyBannerPane" class="MyBannerPane" runat="server"></div>
      <div id="ContentPane" runat="server"></div>
    </div>
    <div id="footer"></div>
  </div>
```

### 6.1 Tipos de temas

DNN admite dos formas diferentes de crear temas usando: archivos HTML y ASCX. Los archivos HTML ofrecen una forma más fácil de acceder a la

tematización de DNN, ya que el DNN convertirá el archivo HTML en un archivo ASCX y utilizará ese archivo ASCX como tema real.

<b>HTML</b>	<b>ASCX</b>
Tema básico.	Tema avanzado.
Se puede crear usando cualquier editor de HTML.	Más fácil de editar usando Visual Studio Code de Microsoft.
Convertido automáticamente a ASCX cuando se instala. Se omiten las etiquetas <body> y <header>.	Usado como está.
Más fácil de crear. Utiliza tokens de DNN, que representan código que llama a las API de DNN. Los tokens se reemplazan con código durante la conversión a ASCX.	Más poderoso. Permite la personalización del código que llama a las API de DNN.

## 6.2 Contenedor

Un contenedor es similar a una plantilla de diseño donde tiene como alcance los módulos, un contenedor es esencialmente un envoltorio para un módulo y se puede usar para diseñar el contenido o agregar otra funcionalidad.

Un contenedor permite integrar módulos realizados por diferentes desarrolladores, integrándolo dentro del portal web, dando como resultado que este sea interactivo para los administradores del portal web.

Puede utilizar diferentes contenedores en una sola página como también se puede mezclar y combinar un contenedor en cualquier hoja de estilo CSS, así mismo se puede tener un contenido predeterminado que será anulado por paneles o módulos individuales, siempre que esta tenga sus propias definiciones de estilos.

Implementation:

HTML

```
<div id="ContentPane" runat="server"></div>
```

## ASCX

```
<%@ Control AutoEventWireup="false" Explicit="True" Inherits="DotNetNuke.UI. Containers.Container" %>  
<div id="ContentPane" runat="server"></div>
```

### 6.3 Crear un archivo CSS

Una hoja de estilo CCS se puede asociar con una plantilla o anteriormente con un SKIN o contenedor dándole el mismo nombre del archivo y colocando en la misma carpeta, cuando se aplica una plantilla de diseño para contenedor a una página web el archivo CSS asociado se aplica automáticamente.

Los archivos se asocian automáticamente entre sí, ejemplo:

- Home.ascx
- Home.css

Las hojas de estilos CSS se aplican jerárquicamente de forma predeterminada se cargan en el siguiente orden:

El predeterminado para el portal web a partir de la versión 8.0 existe una versión default.css.

“Un tema puede elegir qué versión cargar (*DNN, 2021*)”

1. DefaultCss (~/Resources/Shared/stylesheets/dnndefault/8.00/default.css  
Prioridad 5
2. AdminCss( ~/Portals/\_default/admin.css, prioridad 6) — El CSS de administrador para el sitio web.
3. AdminCss( ~/Portals/\_default/admin.css, prioridad 6) — El CSS de administrador para el sitio web.
4. FeatureCss( ~/Resources/\*/\*.css, prioridad 7): algunos CSS para recursos DNN compartidos, como autocompletar, países/regiones y arrastrar y soltar.
5. IeCss( ~/Portals/\_default/ie.css, prioridad 8) — CSS cargado para manejar peculiaridades en internet explorer.
6. ModuleCss( ~/DesktopModules/\*/module.css, prioridad 10) — El CSS para cada tipo de módulo que aparece en la página.

7. ResourceCss( ~/Resources/\*/\*.css, prioridad 12): algunos CSS para recursos DNN compartidos, como la carga de archivos, la seguridad de la contraseña y el consentimiento de cookies.
8. SkinCss( ~/Portals/\*/Skins/\*/skin.css, prioridad 15) — El CSS maestro para el tema debe llamarse skin.css.
9. SpecificSkinCss (~/Portals/\*/Skins/\*/\*.css, priority 20) — El CSS para una plantilla de diseño específica, debe tener el nombre del archivo CSS, donde coincida con el nombre del archivo.  
home.ascx o home.html y home.css.
10. ContainerCss(~/Portals/ /Containers/ /container.css, prioridad 25) — El CSS maestro para todos los contenedores del tema.
11. SpecificContainerCss(~/Portals/\*/Containers/\*/\*.css, priority 30) — El CSS para una plantilla de diseño específica, debe tener el nombre del archivo CSS, donde coincida con el nombre del archivo.  
home.ascx o home.html y home.css.
12. PortalCss(~/Portals\*/portal.css, prioridad 35): el CSS que los administradores de sitios web pueden editar a través de DNN, útil para crear anulaciones de estilos específicas del sitio.
13. DefaultPriority(prioridad 100): si no se especifica una prioridad, todos los demás archivos CSS se cargan con esta prioridad predeterminada. Esto incluye la hoja de estilo de la página, que se puede configurar en la configuración de una página. El nombre de archivo ingresado en esa configuración será una ruta a un archivo en el sistema de archivos del sitio (por ejemplo home.css, se referirá a ~/Portals/0/home.css una página en el sitio con ID 0). No hay garantía sobre el orden dentro de una sola prioridad, por lo que la hoja de estilo de la página puede cargarse antes y/o después de que otras hojas de estilo se carguen en DefaultPriority.

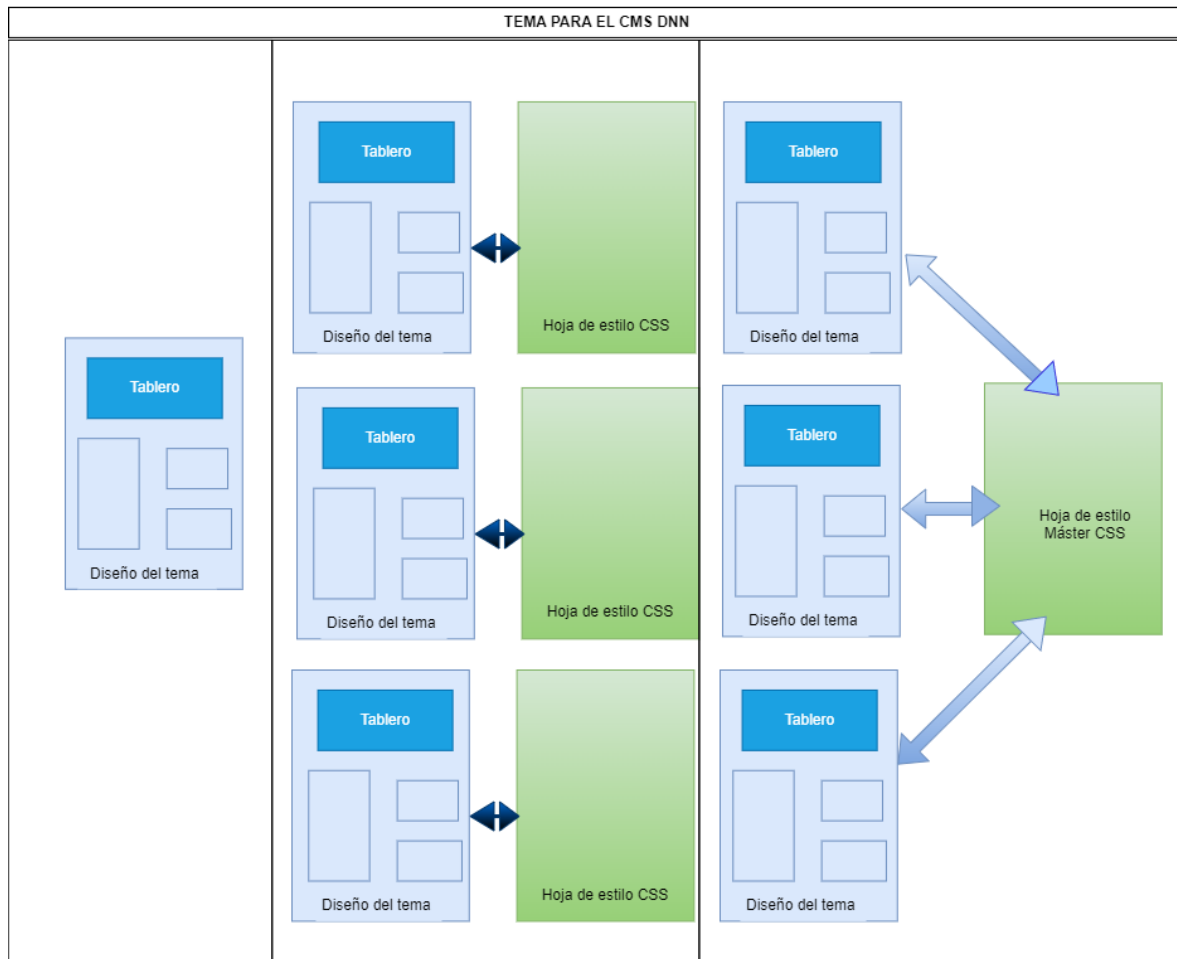
## 6.4 Componentes de un tema

Un tema DNN (anteriormente llamado máscara) define la apariencia del sitio web. Está compuesto por:

Una o más plantillas de diseño (HTML o ASCX).

Una hoja de estilo maestra (CSS) para todas las plantillas de diseño. (opcional)

Una hoja de estilo (CSS) para cada una de las plantillas de diseño. (opcional)



*Ilustración 18. Funcionamiento de un tema*

## 6.5 Creación del Skin para el CMS DNN

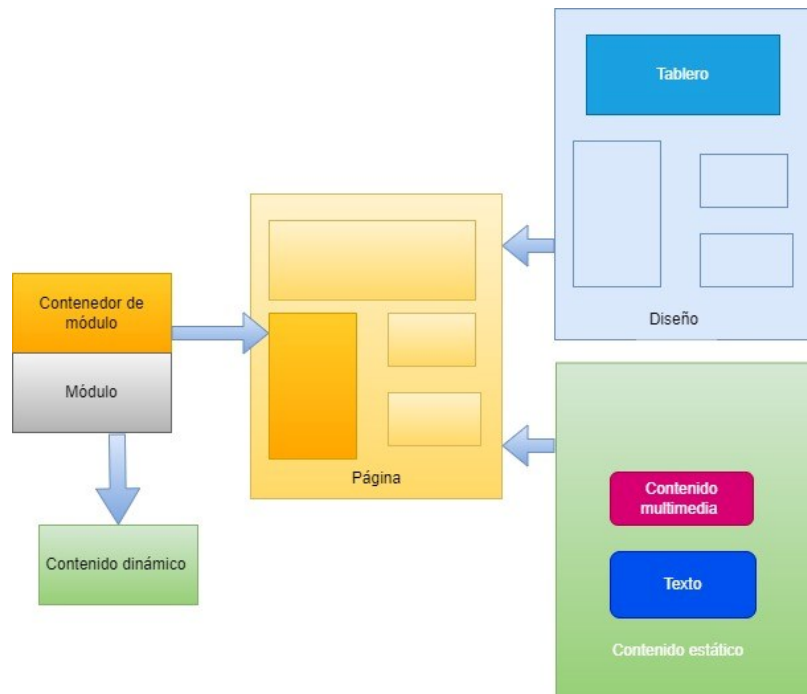
El sistema cuenta con un tema anteriormente llamado Skin, define la apariencia de un sitio web, en nuestro caso migramos nuestro sitio web estático al dinámico, copiando el HTML y reemplazando el texto por módulos propios de DNN.

Dentro del primer punto podemos apreciar el cambio del menú que definimos de manera estática que se definió en función de las páginas que sabíamos que

existían hasta ese momento por uno que se va generando en función de las páginas creadas por el DNN.

```
<!--Header -->
<header role="banner">
  <div id="mainHeader-inner" class="container">
    <div class="clearfix"></div>
    <div class="navbar navbar-default" role="navigation">
      <div id="navbar-top-wrapper">
        <div id="logo">
          <span class="brand">
            <dnn:LOGO runat="server" ID="dnnLOGO" />
          </span>
        </div>
      </div>
      <!-- Brand and toggle get grouped for better mobile display -->
      <div class="navbar-header">
        <button type="button" class="navbar-toggle" data-toggle="collapse"
          data-target=".navbar-collapse">
          <span class="sr-only">Toggle navigation</span>
          <span class="icon-bar"></span>
          <span class="icon-bar"></span>
          <span class="icon-bar"></span>
        </button>
      </div>
      <div id="navbar" class="collapse navbar-collapse pull-right">
        <dnn:MENU ID="MENU" MenuStyle="Menus/MainMenu" runat="server" NodeSelector="*"></dnn:MENU>
      </div>
    </div>
  </div>
</header>
```

## 7. Diseños y Contenedores



*Ilustración 19. Contenedor de módulo*

Un panel es una área o sección de una página web dentro del CMS que muestra contenido estático, contenido dinámico o un objeto de tema estándar.

Una plantilla de diseño define dónde se ubican los paneles en la página web y qué contiene cada panel. Las plantillas de diseño incluidas en un tema son variaciones que el diseñador web puede elegir para cada página del sitio web, como la página de inicio, una galería de fotos o una lista de documentos.

Un contenedor es similar a una plantilla de diseño, excepto que el alcance de un contenedor es un solo un panel, mientras que el alcance de una plantilla de diseño es la página completa. Un contenedor está asociado con un módulo, que puede generar contenido dinámico o realizar otra funcionalidad.

La plantilla de diseño debe contener uno o más paneles. Un contenedor tiene exactamente un panel; los paneles adicionales se ignoran.

Se debe llamar exactamente a uno de los paneles `ContentPane` (no distingue entre mayúsculas y minúsculas).

Cada panel debe tener un nombre que sea único dentro de la página.

- Todos los paneles deben definirse como un control de servidor agregándolos `runat="server"` al elemento.
- Un panel puede ser uno de los siguientes elementos HTML: `<td>` (celdas de tabla), `<div>`, `<p>` y `<span>`.

Los diseños y contenedores pueden contener un objeto de tema, que puede ser un componente típico de una página web, como la barra de menú, el aviso de derechos de autor, los enlaces de inicio de sesión/registro, el enlace de privacidad, el enlace de términos de servicio y el cuadro de búsqueda. Los objetos de tema se insertan en HTML o ASCX como tokens.

## 7.1 Contenedor y diseño alternativo

Este es el Diseño y/o Contenedor que se cargará cuando surja un error en uno de los archivos ascx del Diseño o Contenedor asignado. Normalmente, esto solo debería suceder durante la etapa de desarrollo de un tema o después de una actualización de DNN, cuando el tema existente usa código heredado u objetos de tema. Esta configuración se puede encontrar en el archivo `DotNetNuke.config` en la raíz de la instalación de DNN. El XML debe modificarse para cambiar el aspecto alternativo:

```
<skinningdefaults>  
  <skininfo folder="/Xcillion/" default="Inner.ascx" admindefault="Admin.ascx" />  
  <containerinfo folder="/Xcillion/" default="Title_h2.ascx" admindefault="Title_h2.ascx" />  
</skinningdefaults>
```

Normalmente, solo necesita cambiar esto cuando desinstala una máscara predeterminada del DNN más antigua.

## 7.2 Diseño de portal y contenedor

El Diseño que se establezca para el Portal se aplicará a todas las páginas a menos que anule esto a nivel de página, lo mismo ocurre con el Contenedor del Portal, se aplicará a todos los Módulos del Portal.

### **7.3 Diseño de página y contenedor**

El diseño y/o el contenedor que establezca para una página anularán el diseño y el contenedor de los portales.

#### **NOTA:**

Las páginas secundarias no heredan automáticamente esta configuración. Puede copiar esto manualmente en todas las páginas secundarias, pero las nuevas páginas secundarias no heredarán el diseño de página o el Contenedor.

#### **7.3.1 Contenedor de módulo**

En la configuración del módulo, podemos establecer un contenedor específico para ese módulo. Esto anulará la configuración anterior, pero solo para ese módulo.

#### **7.3.2 Contenedor de panel**

En algunos casos, todos los módulos dentro de un Panel determinado en el diseño necesitan un Contenedor específico. En ese caso, para facilitar la vida de un Administrador del sitio, podemos configurar el Contenedor predeterminado para los Módulos colocados en ese Panel. Esto anulará la configuración del portal y la página, pero no de un módulo. Cualquier módulo colocado en ese Panel cargará el Contenedor de Panel asignado, a menos que esto se haya establecido explícitamente en la configuración del Módulo. No podemos realizar este cambio en la interfaz de administración de DNN. En su lugar, cambiaremos esto en la maquetación.ascx.

### **7.4 Hojas de estilo en cascada (CSS)**

Las hojas de estilo en cascada (archivos CSS) se asocian automáticamente con diseños y contenedores, si están en la misma carpeta y comparten el mismo nombre de archivo.

Paginas.css está asociado con Paginas.ascx

Home.css está asociado con Home.html

- Cualquier archivo CSS asociado con una plantilla de diseño define cómo se muestran los elementos de toda la página.
- Cualquier archivo CSS asociado con un contenedor define cómo se muestra el contenido dinámico dentro de un solo panel de la página web. Esto asegura un diseño unificado en toda la página, incluso si los distintos módulos en la misma página provienen de diferentes creadores.

Si los archivos CSS no están asociados explícitamente con los diseños y contenedores, se utiliza el archivo CSS maestro del sitio web. Si no hay archivos CSS asociados con el sitio web, se utiliza el archivo CSS maestro del host.

Los diseños y contenedores pueden compartir varios archivos CSS o un único archivo CSS maestro.

Cuando se aplica una plantilla de diseño a una página web, cualquier CSS asociado se agrega automáticamente a la página web. Del mismo modo, cuando se aplica un contenedor a un panel, cualquier CSS asociado se agrega automáticamente al panel.

El CMS puede contener una versión de Bootstrap que puede provocar que, si estas clases no tienen el mismo nombre, debemos renombrarlos con la finalidad de que no entre en conflicto, provocando que algunas funcionalidades tengan un comportamiento no esperado.

Con la siguiente configuración ya podríamos generar el portal web, solo bastaría con subir todo el proyecto a una carpeta, debido a que este fue referenciada a enlace relativo no habría problemas al apuntar a un sitio web en vez de un proyecto local.

## 7.5 Crear un XML de tipo de documento

Un diseño destinado a ser visto usando HTML5 no se verá correcto si se representa usando la especificación XHTML o HTML4. Podemos forzar que el tema se represente en XHTML o HTML5 creando un archivo DocType XML (skin.doctype.xml), que se aplica a todos los diseños del tema. Además, los temas pueden contener un archivo DocType separado para cualquier plantilla de diseño individual para anular la declaración DocType del tema.

Ingresamos el código apropiado en un archivo DocType.

```
HTML5
<SkinDocType>
  <![CDATA[<!DOCTYPE html]]>
</SkinDocType>

XHTML estricto
<SkinDocType>
  <![CDATA [<!DOCTYPE html
    PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "https://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">]]>
</SkinDocType>
```

Las hojas de estilo y de JavaScript se agregan en los archivos ascx como lo haríamos en un archivo HTML

```
<script src="/portals/_default/skins/tema_ccyt/js/jquery1.14.js"></script>
```

```
<link rel="stylesheet" href="css/font-awesome.min.css">
```

Se recomienda que si varios de estos archivos se ocupan de manera recurrente en diferentes archivos ascx se puede agregar una página común y así tener una mejor estructuración.

Portals/\_default/Skins/tema\_ccyt/common/AddFiles.ascx

La forma de agregar a los diferentes archivos ascx es similar a como agregamos las bibliotecas de c mediante la siguiente línea de código

```
#Include file="Common/AddFiles.ascx"
```

## 7.6 Empacar la extensión

Para facilitar la distribución e instalación, los componentes de una extensión (tema o módulo) se pueden agrupar en un paquete. Un paquete de extensión es simplemente un archivo zip que contiene todos los archivos requeridos por su extensión. La parte más importante del proceso es crear el Manifiesto DNN, que proporciona la información requerida por el instalador, como las ubicaciones de destino de los archivos.

Si usamos las plantillas del DNN y compilamos en Visual Studio, la compilación de lanzamiento crea el archivo zip.

**Nota:** El asistente de paquetes, accesible a través del Creador de módulos DNN o la página de Extensiones, actualmente no puede crear paquetes para módulos MVC o SPA.

### 7.6.1 Pasos

Poner los archivos en carpetas.

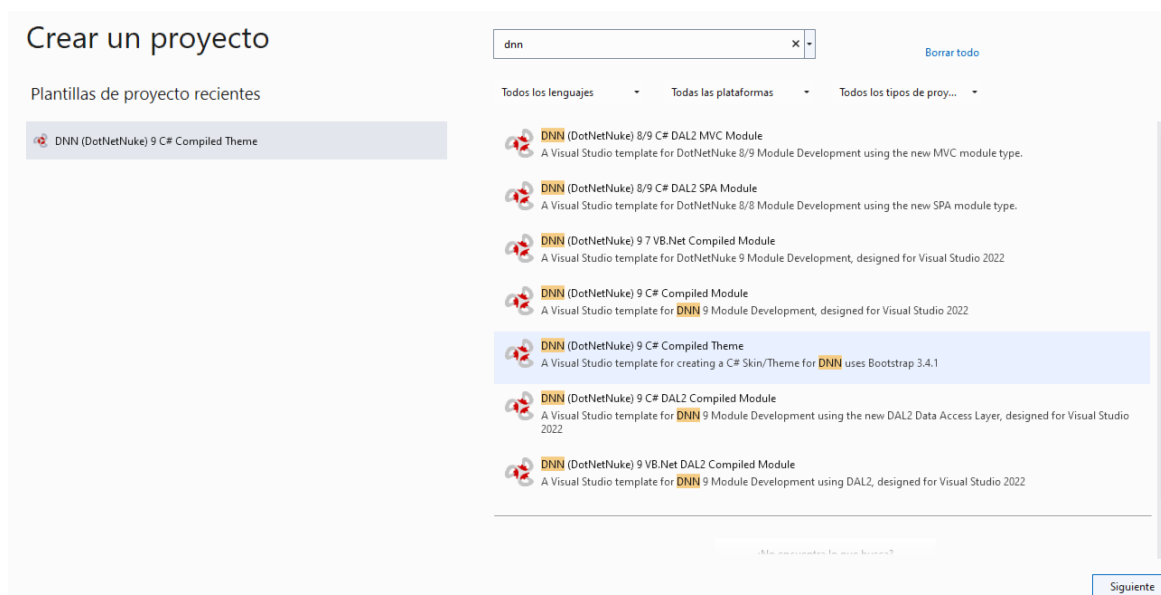
- Archivos que son comunes entre todos los paquetes:
  - (Opcional) Licencia.txt  
Muestra al usuario durante la instalación del paquete.
  - (Opcional) Notas\_version.txt  
Enumera los cambios para la versión actual del paquete y también se proporciona durante la instalación.
- Archivos incluidos en los paquetes de módulos MVC y SPA:
  - Requerido
    - Las vistas (.cshtml o .vbhtml) contienen el marcado necesario para representar la interfaz de usuario de su módulo.
    - El archivo de manifiesto (.dnn) contiene la información de definición del módulo necesaria para instalar el módulo.
    - Los ensamblajes (.dll) son el código del módulo compilado y las bibliotecas de referencia de terceros. Los proyectos de WSP no tendrán un ensamblaje para el módulo compilado, pero aún pueden incluir bibliotecas de referencia de terceros.

- Los scripts SQL (.sqldataprovider) son el código necesario para crear o actualizar los objetos de la base de datos de su módulo.
- Opcional
- Los archivos de recursos (.resx) contienen cadenas de localización.
- Los archivos JavaScript (.js) contienen código utilizado para la lógica del lado del cliente.
- Las hojas de estilo (.css) contienen los estilos personalizados que necesitamos en nuestro módulo.
- Los archivos de texto (.txt) incluyen los archivos release.txt y license.txt que se muestran durante la instalación del módulo.

Sugerencia: la licencia y las notas de la versión son archivos HTML, por lo que puede incluir ofertas especiales, como una llamada a los métodos y otros detalles.

Debemos:

1. Incluir el número de versión de la extensión en las notas de la versión.
2. Cree el manifiesto DNN .
3. Comprimir nuestros archivos, incluido el manifiesto DNN en la carpeta raíz.



*Ilustración 20. Compilar tema DNN*

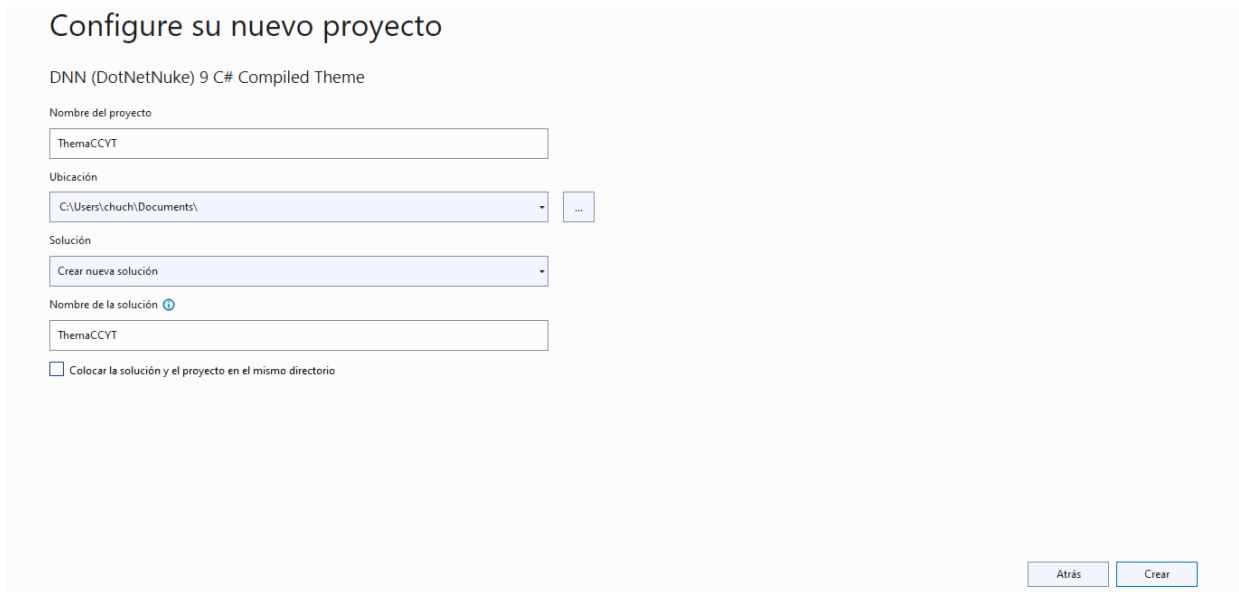


Ilustración 21. Ubicación tema DNN

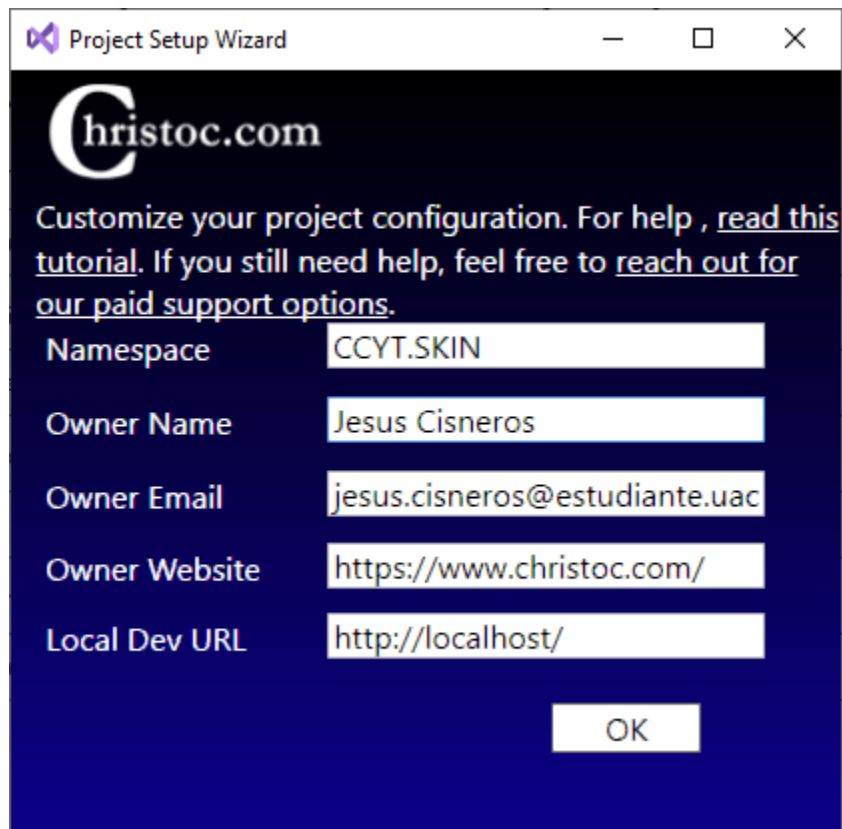
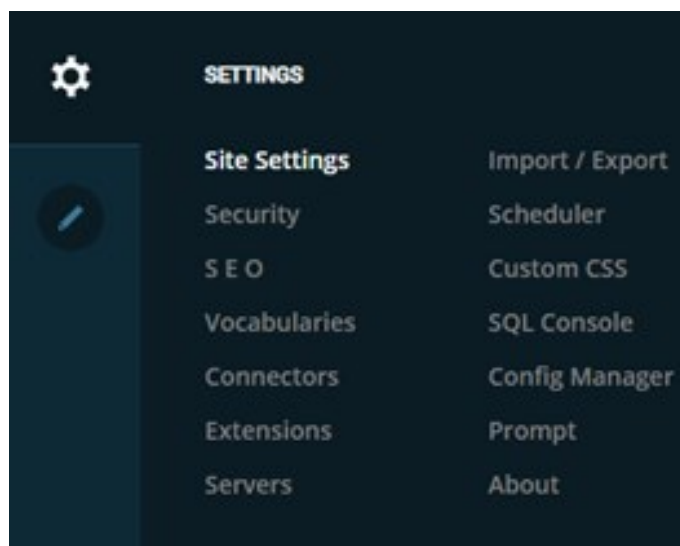


Ilustración 22. Configurar skin DNN

## 8. Administración del CMS DNN

### 8.1 Configuración del DNN

Por default el DNN puede venir con skin, módulos para empezar a trabajar.



*Ilustración 23. Configuración del DNN*

La configuración por default viene en el idioma inglés, recomendamos que se configure en nuestro idioma local en nuestro caso español México, esto con la finalidad de que si posteriormente se desea agregar diferentes idiomas se puedan agregar al mismo sitio, el sistema puede mostrar los diferentes idiomas a los usuarios según nuestra configuración.

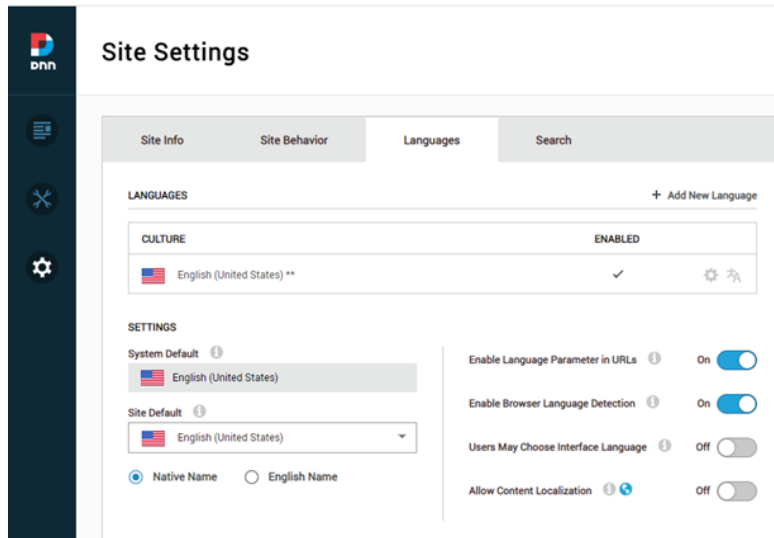


Ilustración 24. Configuración del idioma del DNN

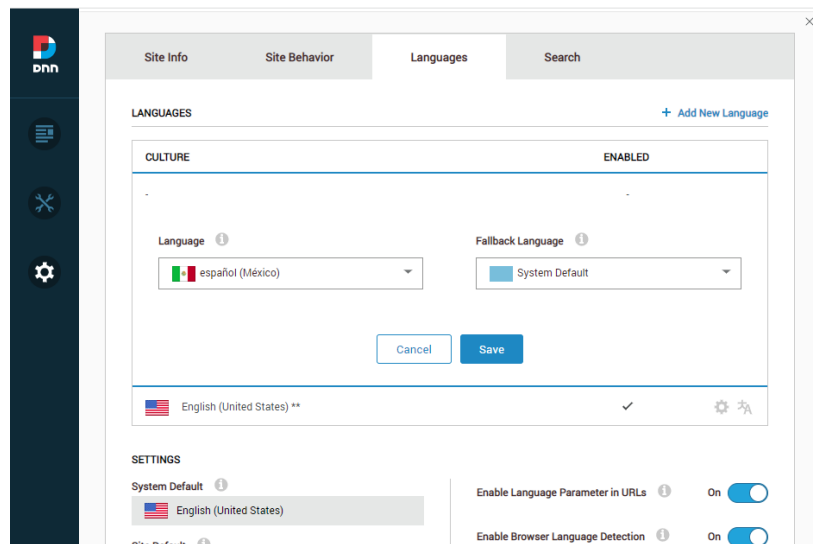
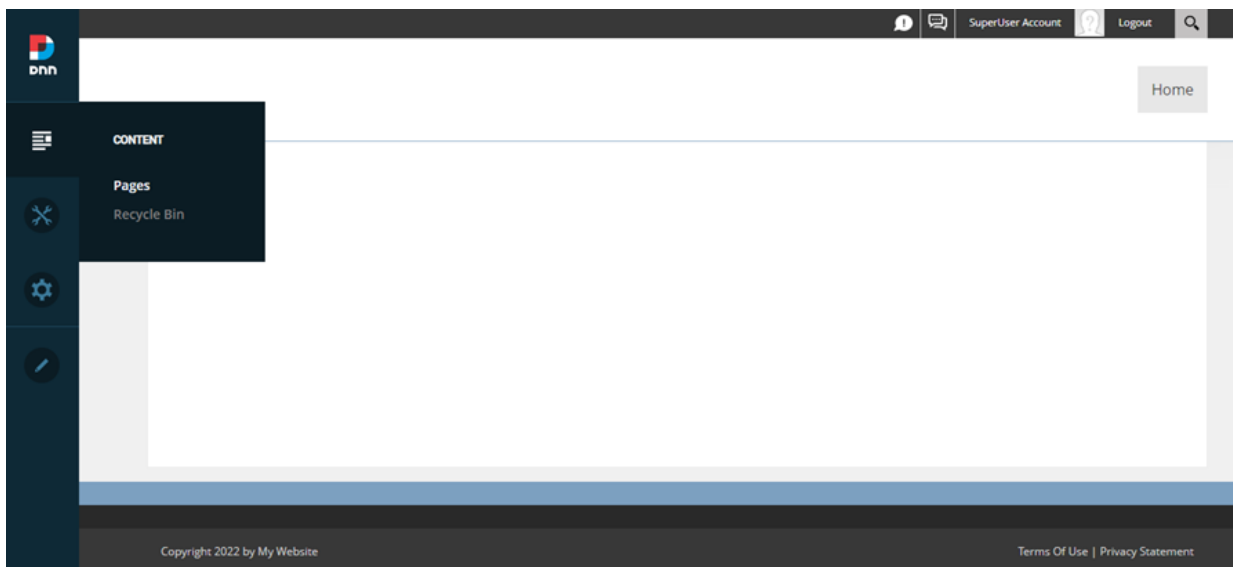


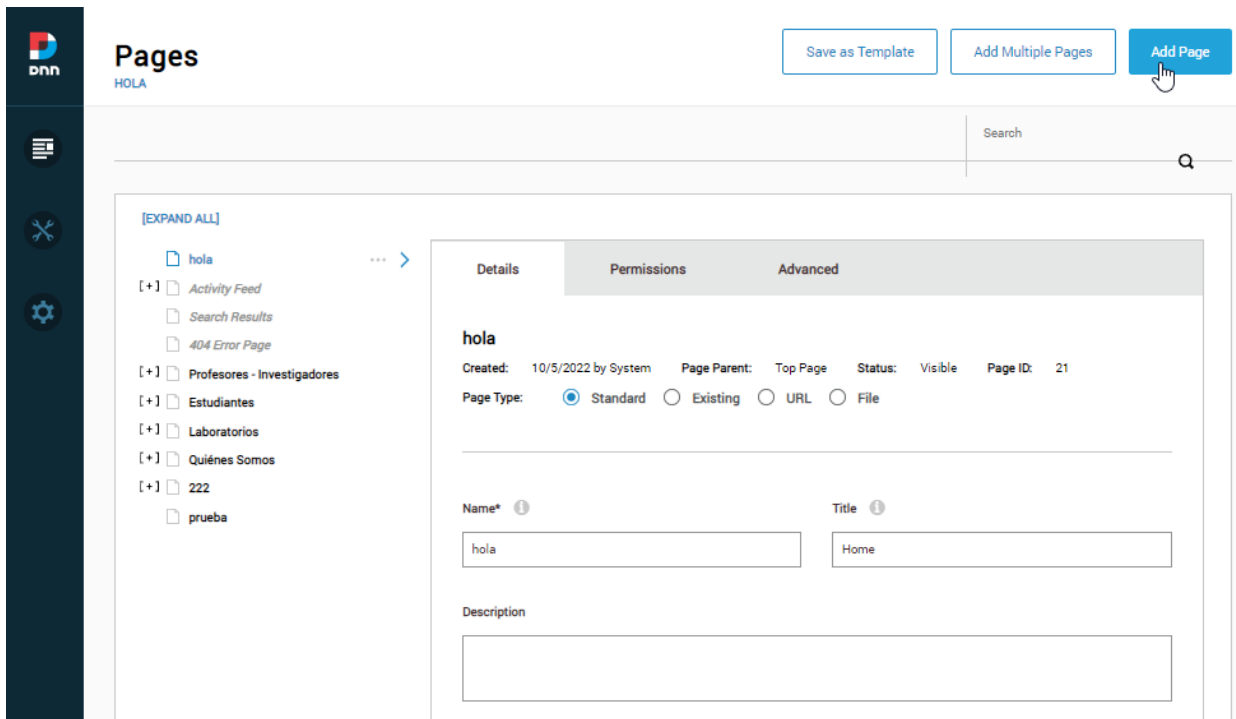
Ilustración 25. Configuración del idioma del DNN parte 2



*Ilustración 26. Página por default del DNN*

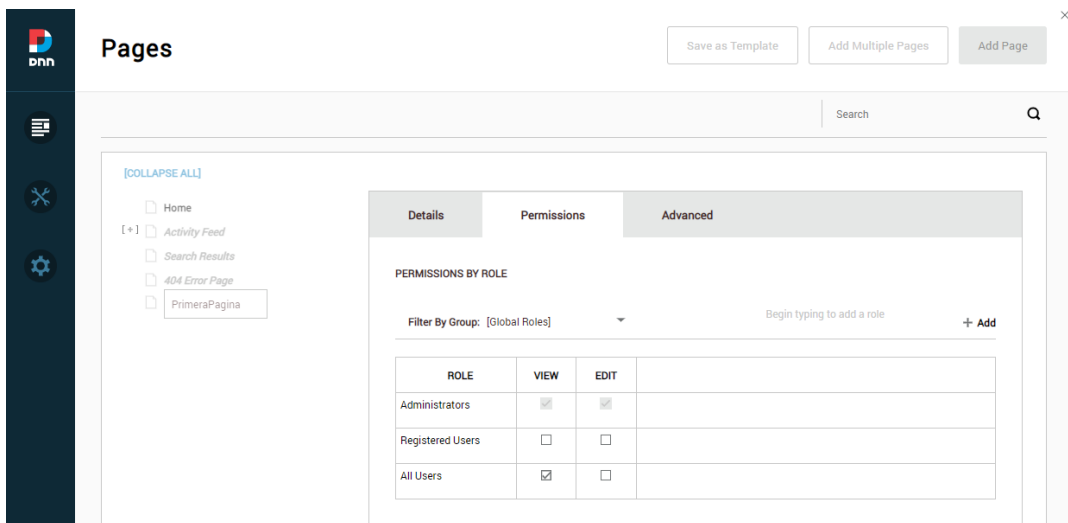
Una vez dentro del panel tendremos la sección de contenido por default el cual nos creara una página en blanco o con el tema por default, en esta versión v.09.10.02 del DNN, el panel de control se encuentra en la parte inferior izquierda donde podemos crear las páginas del portal web o recuperar alguna página o módulo borrado en el apartado de Recycle Bin.

Podemos crear una página, como también es posible crear múltiples páginas, a continuación, se muestra el recorrido para crear una sola página, debemos seguir el asistente para empezar a llenar los datos.



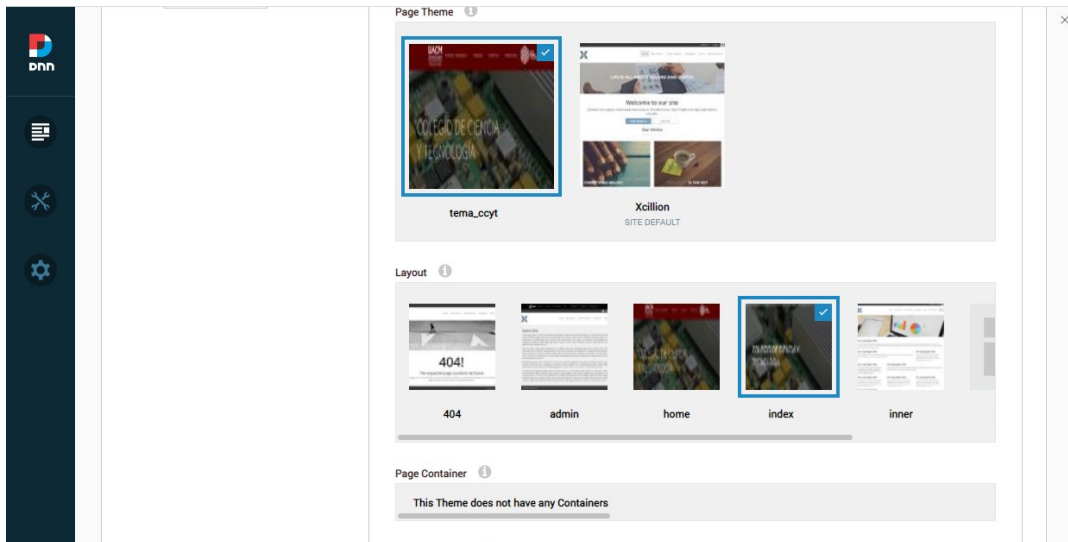
*Ilustración 27. Creación de una página*

La siguiente pestaña nos indica los permisos de la página de quienes pueden editarlo y visualizarlo, ya que pueden ser mostrados a los usuarios con acceso al panel o de manera pública.



*Ilustración 28. Permiso de las páginas*

Elegir plantilla para la página en la pestaña de avanzado procedemos a elegir nuestro skin este puede contener varias plantillas en la que nosotros podemos agregar módulos en la sección ya preestablecida anteriormente.



*Ilustración 29. Agregar plantilla*

La siguiente tarea es copiar el contenido que se va a mantener estático, cuidando que la colocación sea lo más parecido a la que se tenía anteriormente, recordando que se tienen nuevos elementos propios del gestor, como por ejemplo el JQuery y la propia versión de Bootstrap.

Al iniciar sesión como administrador del portal es posible que los elementos se muevan de lugar debido a que mostrará elementos del DOM que se mantenían ocultos, esta segunda vista solo será mostrada para los administradores del sitio web, para esta parte lo que se debe de cuidar es la parte del funcionamiento del CMS:

- La creación correcta del menú, tanto visualmente, como al agregar una página aparezca correctamente y nos dirija al sitio web.
- Agregar módulos
- La edición de módulos
- Borrar módulo

La primera página con la que se empezó a trabajar la llamamos Home, dentro de esta página tenemos al inicio la importación de nuestras bibliotecas tal y como lo haríamos en un lenguaje de programación, indicando al inicio el tipo de lenguaje de programación con el que deseamos trabajar, recordando que para este CMS en particular soporta los lenguajes de programación Visual Basic y C#.

```
<%@ Control Language="vb" AutoEventWireup="false" Explicit="True" Inherits="DotNetNuke.UI.Skins.Skin" %>
<%@ Register TagPrefix="dnn" TagName="STYLES" Src="~/Admin/Skins/Styles.ascx" %>
<%@ Register TagPrefix="dnn" TagName="CURRENTDATE" Src="~/Admin/Skins/CurrentDate.ascx" %>
<%@ Register TagPrefix="dnn" TagName="LANGUAGE" Src="~/Admin/Skins/Language.ascx" %>
<%@ Register TagPrefix="dnn" TagName="LOGO" Src="~/Admin/Skins/Logo.ascx" %>
<%@ Register TagPrefix="dnn" TagName="SEARCH" Src="~/Admin/Skins/Search.ascx" %>
<%@ Register TagPrefix="dnn" TagName="USER" Src="~/Admin/Skins/User.ascx" %>
<%@ Register TagPrefix="dnn" TagName="LOGIN" Src="~/Admin/Skins/Login.ascx" %>
<%@ Register TagPrefix="dnn" TagName="PRIVACY" Src="~/Admin/Skins/Privacy.ascx" %>
<%@ Register TagPrefix="dnn" TagName="TERMS" Src="~/Admin/Skins/Terms.ascx" %>
<%@ Register TagPrefix="dnn" TagName="BREADCRUMB" Src="~/Admin/Skins/BreadCrumb.ascx" %>
<%@ Register TagPrefix="dnn" TagName="COPYRIGHT" Src="~/Admin/Skins/Copyright.ascx" %>
<%@ Register TagPrefix="dnn" TagName="LINKTOMOBILE" Src="~/Admin/Skins/LinkToMobileSite.ascx" %>
<%@ Register TagPrefix="dnn" TagName="META" Src="~/Admin/Skins/Meta.ascx" %>
<%@ Register TagPrefix="dnn" TagName="MENU" Src="~/DesktopModules/DDRMenu/Menu.ascx" %>
<%@ Register TagPrefix="dnn" Namespace="DotNetNuke.Web.Client.ClientResourceManagement"
Assembly="DotNetNuke.Web.Client" %>
<%@ Register TagPrefix="dnn" TagName="jQuery" Src="~/Admin/Skins/jquery.ascx" %>
```

La segunda parte nos aseguramos de que no usen internet explorer debido a que este perdió soporte por parte de la empresa de Microsoft.

```
<[if lt IE 9]>
  <script src="https://cdnjs.cloudflare.com/ajax/libs/html5shiv/3.7.2/html5shiv.min.js"></script>
  <script src="https://oss.maxcdn.com/respond/1.4.2/respond.min.js"></script>
<[/endif]>
```

Los siguientes elementos corresponden a los elementos propios del gestor del DNN, combinados con los elementos propios, donde los colocaremos según nuestro diseño, mandamos a llamar las clases css y del sistema. Componente de skin DNN

```
<!-- UserControlPanel -->
  <div id="topHeader">
    <div class="container">

      <div class="row">
        <div class="col-md-12">
          <div class="language">
            <dnn:LANGUAGE runat="server" ID="LANGUAGE1" ShowMenu="False"
              ShowLinks="True" />
          </div>
          <div class="search hidden-xs">
            <dnn:SEARCH ID="dnnSearch" runat="server" ShowSite="false" ShowWeb="false"
              EnableTheming="true" Submit="Search" CssClass="SearchButton" />
          </div>
          <!-- search action for Search function on small devices -->
          <a id="search-action" aria-label="Search"></a>
          <div id="login" class="pull-right">
            <dnn:LOGIN ID="dnnLogin" CssClass="LoginLink" runat="server"
              LegacyMode="false" />
            <dnn:USER ID="dnnUser" runat="server" LegacyMode="false" />
          </div>
        </div>
      </div>
    </div>
  </div>
<!-- UserControlPanel -->
  <div id="topHeader">
    <div class="container">

      <div class="row">
        <div class="col-md-12">
          <div class="language">
            <dnn:LANGUAGE runat="server" ID="LANGUAGE1" ShowMenu="False"
              ShowLinks="True" />
          </div>
          <div class="search hidden-xs">
```

```

        <dnn:SEARCH ID="dnnSearch" runat="server" ShowSite="false" ShowWeb="false"
            EnableTheming="true" Submit="Search" CssClass="SearchButton" />
    </div>
    <!-- search action for Search function on small devices -->
    <a id="search-action" aria-label="Search"></a>
    <div id="login" class="pull-right">
        <dnn:LOGIN ID="dnnLogin" CssClass="LoginLink" runat="server"
            LegacyMode="false" />
        <dnn:USER ID="dnnUser" runat="server" LegacyMode="false" />
    </div>
</div>
</div>
</div>
</div>
<!-- UserControlPanel -->
<div id="topHeader">
    <div class="container">

        <div class="row">
            <div class="col-md-12">
                <div class="language">
                    <dnn:LANGUAGE runat="server" ID="LANGUAGE1" ShowMenu="False"
                        ShowLinks="True" />
                </div>
                <div class="search hidden-xs">
                    <dnn:SEARCH ID="dnnSearch" runat="server" ShowSite="false" ShowWeb="false"
                        EnableTheming="true" Submit="Search" CssClass="SearchButton" />
                </div>
                <!-- search action for Search function on small devices -->
                <a id="search-action" aria-label="Search"></a>
                <div id="login" class="pull-right">
                    <dnn:LOGIN ID="dnnLogin" CssClass="LoginLink" runat="server"
                        LegacyMode="false" />
                    <dnn:USER ID="dnnUser" runat="server" LegacyMode="false" />
                </div>
            </div>
        </div>
    </div>
</div>
</div>
<!-- UserControlPanel -->
<div id="topHeader">
    <div class="container">

        <div class="row">
            <div class="col-md-12">
                <div class="language">
                    <dnn:LANGUAGE runat="server" ID="LANGUAGE1" ShowMenu="False"
                        ShowLinks="True" />
                </div>
                <div class="search hidden-xs">
                    <dnn:SEARCH ID="dnnSearch" runat="server" ShowSite="false" ShowWeb="false"
                        EnableTheming="true" Submit="Search" CssClass="SearchButton" />
                </div>
                <!-- search action for Search function on small devices -->

```

```

        <a id="search-action" aria-label="Search"></a>
        <div id="login" class="pull-right">
            <dnn:LOGIN ID="dnnLogin" CssClass="LoginLink" runat="server"
                LegacyMode="false" />
            <dnn:USER ID="dnnUser" runat="server" LegacyMode="false" />
        </div>
    </div>
</div>
</div>
</div>
<!-- UserControlPanel -->
<div id="topHeader">
    <div class="container">

        <div class="row">
            <div class="col-md-12">
                <div class="language">
                    <dnn:LANGUAGE runat="server" ID="LANGUAGE1" ShowMenu="False"
                        ShowLinks="True" />
                </div>
                <div class="search hidden-xs">
                    <dnn:SEARCH ID="dnnSearch" runat="server" ShowSite="false" ShowWeb="false"
                        EnableTheming="true" Submit="Search" CssClass="SearchButton" />
                </div>
                <!-- search action for Search function on small devices -->
                <a id="search-action" aria-label="Search"></a>
                <div id="login" class="pull-right">
                    <dnn:LOGIN ID="dnnLogin" CssClass="LoginLink" runat="server"
                        LegacyMode="false" />
                    <dnn:USER ID="dnnUser" runat="server" LegacyMode="false" />
                </div>
            </div>
        </div>
    </div>
</div>
</div>
<!-- UserControlPanel -->
<div id="topHeader">
    <div class="container">

        <div class="row">
            <div class="col-md-12">
                <div class="language">
                    <dnn:LANGUAGE runat="server" ID="LANGUAGE1" ShowMenu="False"
                        ShowLinks="True" />
                </div>
                <div class="search hidden-xs">
                    <dnn:SEARCH ID="dnnSearch" runat="server" ShowSite="false" ShowWeb="false"
                        EnableTheming="true" Submit="Search" CssClass="SearchButton" />
                </div>
                <!-- search action for Search function on small devices -->
                <a id="search-action" aria-label="Search"></a>
                <div id="login" class="pull-right">
                    <dnn:LOGIN ID="dnnLogin" CssClass="LoginLink" runat="server"
                        LegacyMode="false" />

```

```

        <dnn:USER ID="dnnUser" runat="server" LegacyMode="false" />
    </div>
</div>
</div>
</div>
</div>
</div>
<!--Header -->
<header role="banner">
    <div id="mainHeader-inner" class="container">
        <div class="clearfix"></div>
        <div class="navbar navbar-default" role="navigation">
            <div id="navbar-top-wrapper">
                <div id="logo">
                    <span class="brand">
                        <dnn:LOGO runat="server" ID="dnnLOGO" />
                    </span>
                </div>
            </div>
            <!-- Brand and toggle get grouped for better mobile display -->
            <div class="navbar-header">
                <button type="button" class="navbar-toggle" data-toggle="collapse"
                    data-target=".navbar-collapse">
                    <span class="sr-only">Toggle navigation</span>
                    <span class="icon-bar"></span>
                    <span class="icon-bar"></span>
                    <span class="icon-bar"></span>
                </button>
            </div>
            <div id="navbar" class="collapse navbar-collapse pull-right">
                <dnn:MENU ID="MENU" MenuStyle="Menus/MainMenu" runat="server" NodeSelector="*"></dnn:MENU>
            </div>
        </div>
    </div>
</header>

<!-- Page Content -->
<div class="container">
    <main role="main">
        <div class="row dnnpane">
            <div id="HeaderPane" class="col-md-12 headerPane" runat="server"></div>

        </div>
        <div id="mainContent-inner">
            <div class="row dnnpane">
                <div id="ContentPane" class="col-md-12 contentPane" runat="server"></div>
                <p>donde estoy</p>
            </div>

            <div class="row dnnpane">
                <div id="P1_75_1" class="col-md-8 spacingTop" runat="server"></div>
                <div id="P1_25_2" class="col-md-4 spacingTop" runat="server"></div>
                <p>pepe</p>
            </div>
        </div>
    </main>
</div>

```

```
<div class="row dnnpane">
  <div id="P2_25_1" class="col-md-4 spacingTop" runat="server"></div>
  <div id="P2_75_2" class="col-md-8 spacingTop" runat="server"></div>
</div>

<div class="row dnnpane">
  <p>caca</p>
  <div id="P3_33_1" class="col-md-4 spacingTop" runat="server"></div>
  <div id="P3_33_2" class="col-md-4 spacingTop" runat="server"></div>
  <div id="P3_33_3" class="col-md-4 spacingTop" runat="server"></div>
</div>

<div class="row dnnpane">
  <div id="ContentPanelLower" class="col-md-12 contentPane spacingTop"
    runat="server"></div>
</div>
</div><!--
  /.mainContent-inner -->
</main>
<!-- /.mainContent -->
</div>
<!-- /.container -->

<!-- Footer -->
```

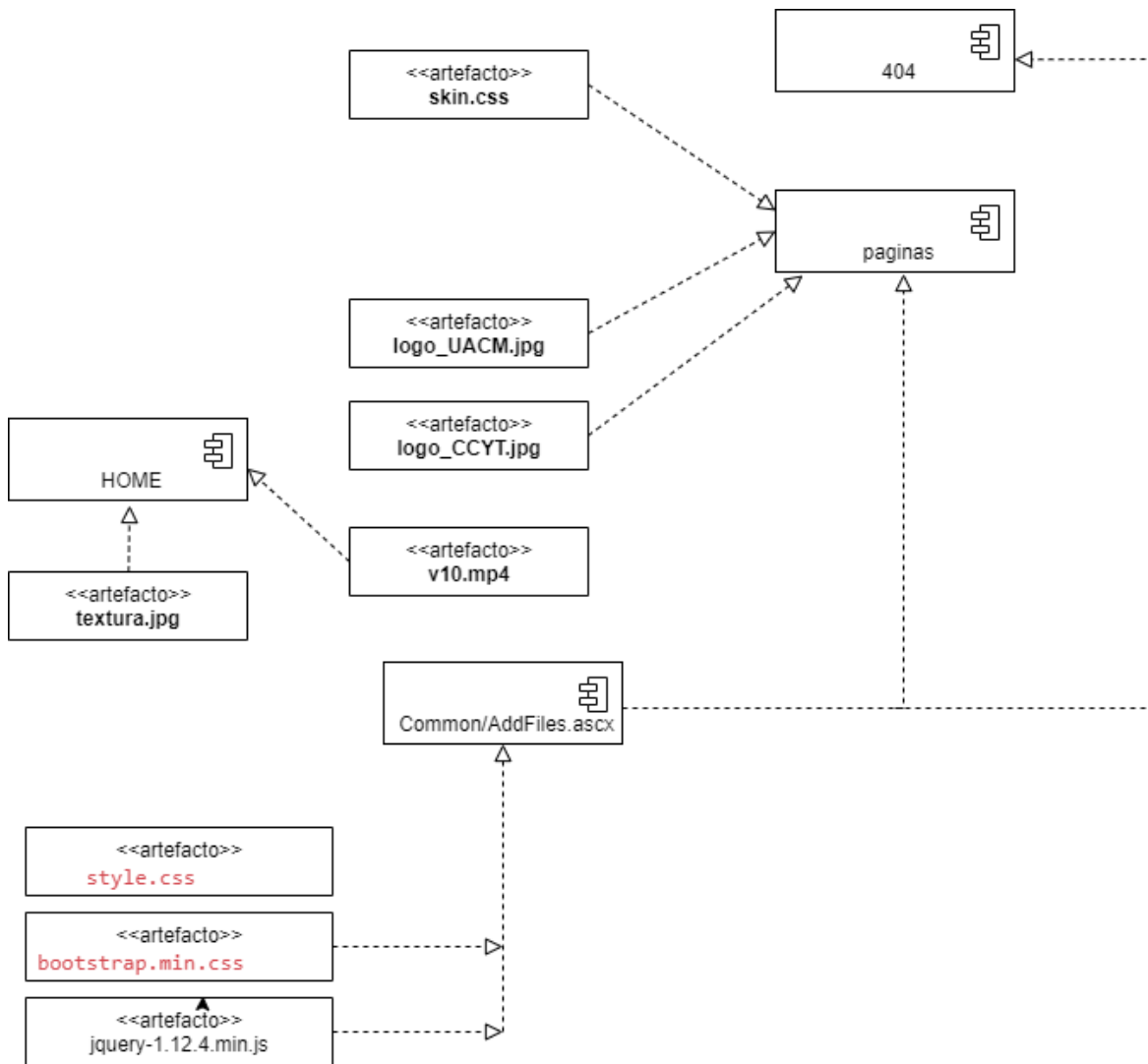


Ilustración 30. Componente de skin DNN

## 9. Seguridad

### 9.1 El JSON Web Token (JWT)

Es un formato de datos de estándar abierto (IETF RFC 7519) que es compacto, autónomo y seguro. Está diseñado para pasar información donde el espacio es limitado, como encabezados HTTP y consultas de URI.

Compacto. Debido a que el JWT se compone de objetos de notación de objetos JavaScript (JSON) codificados, es lo suficientemente compacto como para enviarse a través de una consulta de URL, un parámetro POST o un encabezado

HTTP. Los objetos JSON son más simples y compactos que las aserciones del lenguaje de marcado de aserción de seguridad (SAML), que utilizan XML. Debido a su tamaño más pequeño, también se puede transmitir más rápido.

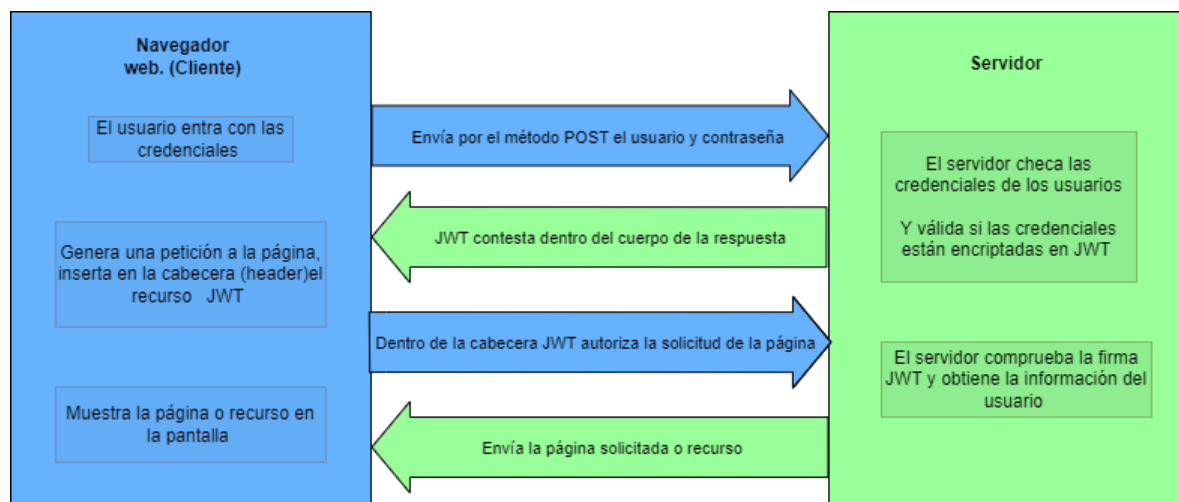
Autónomo. El JWT puede contener toda la información requerida sobre el usuario y, por lo tanto, evita consultar la base de datos más de una vez.

Seguro. El JWT se puede firmar digitalmente con uno de los siguientes métodos:  
Algoritmo HMAC, usando un secreto.

Algoritmo RSA, utilizando un par de claves pública/privada

El JWT es ideal para un portal web o una aplicación web que no usan cookies para tomar la sesión. En una implementación estándar de formularios web, cuando un usuario visita el portal web y recibe un token de cookie/sesión devuelta por el navegador en cada solicitud posterior al sitio, para evitar verificar las credenciales del usuario en cada solicitud. El JWT simplemente reemplaza la cookie con un token de transferencia más pequeño y rápido.

**NOTA:** El proveedor de autenticación JWT está disponible en productos DNN, sin embargo, debe instalarse y habilitarse por separado. DNN usa JWT solo para autenticación.



*Ilustración 31. JWT*

El usuario inicia sesión con su nombre de usuario y contraseña u otras credenciales de seguridad. El navegador o la aplicación cliente envía una solicitud POST con las credenciales del usuario, que se envían a través de una conexión HTTPS.

Las credenciales del usuario se comparan con la base de datos de inicio de sesión. Si es válido, el servidor crea y cifra un JWT de acceso, que se almacena en el cuerpo de la respuesta.

Cuando el usuario solicita una página, el navegador o la aplicación cliente almacena el JWT de acceso dentro de la autorización de la solicitud.

El servidor verifica la firma JWT y extrae la información del usuario de la carga útil de JWT.

La página o recurso solicitado se envía al cliente.

La configuración de privacidad se encuentra en la configuración del sitio.