**PublicCMS**

**1. Beskrivning av projektet**

PublicCMS är ett innehållshanteringssystem för webbplatser (Content Management System).

Syftet är att göra det enkelt att skapa och underhålla en webbplats utan djupare tekniska kunskaper.

**2. Tidsöversikt**

Projektet startade 2021-05-17 och avslutades 2021-06-18.

**3. Tekniska krav**

PublicCMS kan köras på Azure och på serverlösningar som stödjer ASP .NET CORE 5. Vi har valt Asp.Net Core 5 eftersom det är en stabil och säker plattform med stor flexibilitet när det gäller servermiljö för hostning.

Som huvuddatabas för innehållet har vi valt att använda en dokumentdatabas. Detta ger en god flexibilitet för ett system som detta. Applikationen använder två databaser. Huvuddatabaslösningen för applikationen är Mongo DB (på Azure innebär det Cosmos DB med Mongo DB API som gränssnitt). För autentisering används en Sql-databas som är förkonfigurerad som en SQLite, men kan enkelt bytas till en SQL server om man så vill.

Utöver de vanliga NuGet-paketen innehåller PublicCMS paketen Entity Framework, SQLite, Microsoft Identity, Mongo DB Driver.

**4. Applikationsstruktur**

Applikationen använder sig av Razor Pages och består av två delar; en huvuddel och en Microservice del som hanterar besöksräknarfunktionalitet. Huvudprojektet heter PublicCMS och Microservicen heter VisitorCounterAPI. Dessa installeras på separata instanser. När en besökare besöker en webbsida skickas automatiskt ett anrop till siteräknar API: et (endast icke-inloggade besök räknas). Om besöksräknar API: et inte skulle fungera kommer loggning att ske och webbsidan fortsätter att fungera normalt.

PublicCms stödjer också insticksmoduler som utvecklas helt separat från huvudkoden. Det finns ett interface (IPlugin) som insticksmodulerna behöver implementera. En exempelmodul som visar vädret i Stockholm medföljer kodleveransen.

PublicCMS använder sig av Dependency Injection mönstret och har ett API för att kunna lista webbplatsens sidor som heter SiteMap. API: et kan nås på /api/sitemap.

**5. Installation**

Installationen inleds med att de två solutions som finns inom projektet (PublicCMS och VisitorCounterAPI) publiceras till Azure eller annan plats där det ska hostas via Visual Studio. Innan publicering behöver PublicCMS uppdateras med Connection strängar till Mongo DB och URL till VisitorCounterAPI.

Uppdatering av Connection strängar i fil appsettings.json

* Connection sträng för Identity hittas i gruppen “ConnectionStrings” : “IdentityConnection” (för databas med inloggningsinfo).
* Connection sträng för Mongo DB hittas i gruppen “MongoDB” : “Connectionstring” (databasnamnet finns också angivet där. Denna skapas automatiskt om den inte finns.)
* Connection sträng för API hittas i gruppen “APISettings” : “VisitorsAPI”.

När applikationen startas första gången skapas automatiskt en användare med ett standard lösenord (användarnamn: admin lösenord: Admin123!).

När sajten startats upp kan man direkt gå till adressen <sajtadress>/cms

Här finns hela admin-gränssnittet, och det första steget bör vara att ändra till ett säkert lösenord via admin-menyn.

Nästa steg är att skapa den första sidan under hantera innehåll, och sedan i tabellen med sidor klicka på “Använd som startsida”. Innan detta gjorts kommer sajtens frontend visa “Ingen startsida”.

Det finns en tredje solution i projektet som innehåller ett exempel på en insticksmodul. Denna är vid leveransen förkompilerad och inlagd i mappen Plugins i PublicCMS projektet där den måste ligga i kompilerad form. Om man vill göra ändringar i insticksmodulen får man göra det separat i dess egen solution. I så fall får man sedan kopiera den nya .dll filen till mappen Plugins i huvudprojektet.

**6. Endpoints**

PublicCMS SiteMap API har följande endpoint: \api\sitemap. För VisitorCounterAPI se den bifogade swagger filen VisitorCounterAPISwagger.json som finns i samma mapp som detta dokument.

**7. Funktionalitet**

Inloggningsfunktionalitet

PublicCMS har för närvarande inloggning för en administratör. Inloggningsuppgifterna samt rollen för administratören hanteras av Microsoft Identity biblioteket. Inloggningssidan nås genom att antingen skriva sitens webbaddress/login eller gå till webbaddress/cms. När administratören är inloggad och besöker webbplatsens vanliga sidor, visas en knapp för att kunna gå till adminläget där administratören bland annat kan logga ut. Dessa syns inte för vanliga besökare.

Administratörsfunktionalitet

I adminläget hanteras samtliga funktioner för att skapa och redigera webbplatsen. De kan delas in i följande:

* Hantera innehåll, här kan du skapa och redigera webbplatsens sidor.
* Hantera sajtnavigation, här väljer du vilka sidor och i vilken ordning dessa ska visas i sajtens navigation.
* Välj tema, här kan du välja mellan ett antal färgteman för din webbplats.
* Sajtinställningar, här kan du ställa in namnet på webbplatsen som visas i namnlisten på fönstret/fliken i webbläsaren. Du kan även ställa in den text som ska visas på cookiebannern som användare måste godkänna. Här ställer du även in en URL till din sida för integritetspolicy.
* Visa besökarstatistik, här kan du se hur sidorna på webbplatsen har besökts.
* Ändra lösenord, här ändrar du administratörens lösenord.

Lägga till och redigera sidor

En sida på webbplatsen består av ett namn, en webbadress-del samt ett antal olika delar som bygger upp sidans innehåll. Dels kan du lägga till enskilda enkla delar till exempel textblock, bild och länk. Sedan kan du använda en visuell textredigerare för att kunna anpassa din text som du vill ha den med typsnitt, färg och så vidare. Förutom sidans huvudinnehåll kan du också lägga till dessa delar i sidans sidofält (högra del) och sidans sidfot.

**8. Brister**

Produkten har för närvarande enbart stöd för en administratör.

**9. Förbättringsförslag**

* Hantera fler administratörer.
* Hantera utkast så att administratören enkelt kan ångra ändringar.
* Utökad funktionalitet när det gäller uppladdning av bilder och filer.
* Möjlighet att använda behållarobjekt på sidor (listor, rader och kolumner) för att bättre kunna styra innehållet.
* Förbättrat installationsförfarande där man kan ange sitens namn, adminlösenord med mera.

**10. Summering**

Vi hoppas att detta publiceringssystem ska kunna hjälpa dig kommunicera din verksamhet på din nystartade webbplats. Då systemet har tagits fram på relativt kort tid finns det givetvis utrymme för utökande av funktionalitet och förfinande av befintlig funktionalitet.