

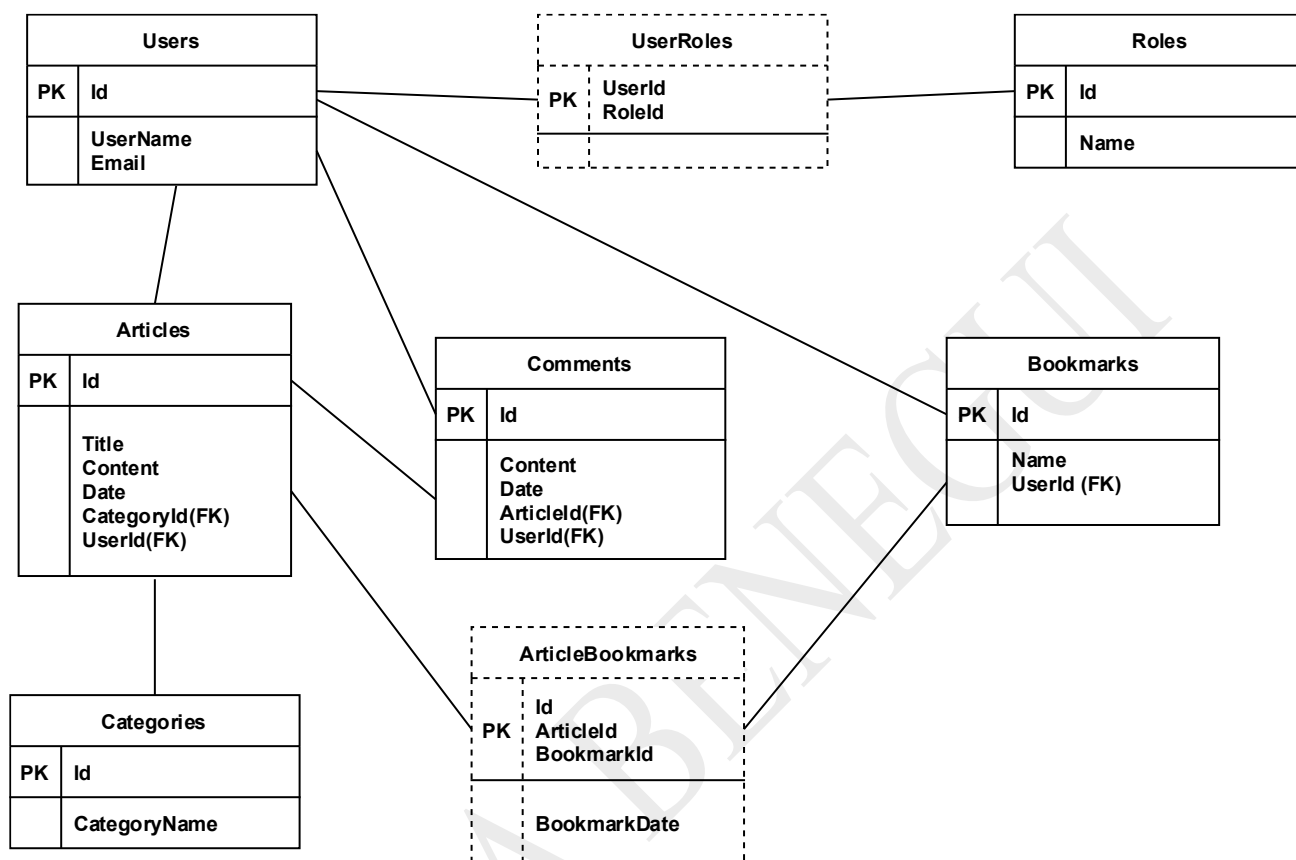
Dezvoltarea Aplicațiilor Web utilizând ASP.NET Core MVC

Laborator 10

EXERCITII:

1. Pornind de la următoarele specificații, să se implementeze posibilitatea de adăugare a articolelor în colecții:
 - Un utilizator își poate crea propriile colecții (Bookmarks).
 - În momentul în care un utilizator își adaugă colecții, acesta o să aibă acces doar la colecțiile pe care le-a creat, existând posibilitatea editării și ștergerii lor.
 - Atât utilizatorii cu rolurile User sau Editor, cât și utilizatorii cu rolul Admin pot crea propriile colecții. Utilizatorii cu rolul User sau Editor o să aibă acces doar la colecțiile create de ei, iar utilizatorii cu rolul Admin o să aibă acces la toate colecțiile existente în platformă.
 - După crearea colecțiilor, utilizatorii își pot adăuga articole în colecții. Articolele pe care le pot adăuga trebuie să existe în platformă. Ei doar selectează un articol și îl adaugă într-o colecție. Utilizatorii nu pot vedea colecțiile altor utilizatori. Fiecare utilizator are acces doar la colecțiile sale.
 - Un articol poate face parte din mai multe colecții, iar o colecție poate conține mai multe articole.

Se consideră Diagrama Conceptuală:



Se modifică, pe rând, clasele *Bookmark*, *ArticleBookmark*, *Article*, *ApplicationUser* și contextul bazei de date *ApplicationDbContext* (VEZI Curs 10).

După modificarea claselor, se realizează o migrație.

După migrație, se vor implementa cerințele de mai sus.

2. Să se implementeze funcționalitatea prin care utilizatorii cu rolul de Admin pot activa sau revoca drepturile celorlalți utilizatori. De asemenea, administratorii pot edita și șterge ceilalți utilizatori din aplicație.

3. Pornind de la noțiunile studiate în cadrul **Cursului 10**, să se adauge în proiect funcționalitatea de afișare paginată a articolelor (**VEZI Curs 10 – Secțiunea “Afișarea paginată”**).
4. Să se integreze editorul de text **Summernote** (**VEZI Curs 10 – Secțiunea “Includerea unui editor de text”**).
5. Să se implementeze motorul de căutare. Articolele pot fi căutate după titlu, conținut sau după conținutul comentariilor asociate acestora (**VEZI Curs 10 – Secțiunea “Funcționalitatea de căutare”**).
6. Fonturile din platformă – pentru alegerea fonturilor se poate utiliza <https://fonts.google.com/>.

1. Se caută sau se aleg fonturile dorite (ex: Roboto, Montserrat, Permanent Marker). Pentru adăugare, se apasă acel “plus” din dreptul fontului. În meniu, în partea dreaptă sus, există opțiunea **“View selected families”**. În această fereastră, în secțiunea **@import**, se copiază secvența de cod din interiorul tagului `<style>` (fără tagul `<style>`) și se adaugă în proiect, în **wwwroot** → **site.css** → **pe prima linie**. După acest pas, se pot utiliza aceste fonturi cu ajutorul atributului `font-family`.

```
font-family: 'Roboto', sans-serif;
font-family: 'Montserrat', sans-serif;
font-family: 'Permanent Marker', cursive;
```

Exemplu:

```
.font-montserrat {
    font-family: 'Montserrat', sans-serif;
}

.font-roboto {
    font-family: 'Roboto', sans-serif;
}

.font-logo {
    font-family: 'Permanent Marker', cursive;
}
```

7. **Stilizare logo** – pentru logo se poate utiliza fontul *Permanent Marker* și se poate stiliza cu ajutorul unei clase (în site.css), după care se folosește clasa în _Layout.cshtml.

De exemplu:

_Layout.cshtml

```
<a class="navbar-brand font-logo" asp-area="" asp-
controller="Articles" asp-action="Index">ArticlesApp</a>
```

wwwroot - site.css

```
.font-logo {
    font-family: 'Permanent Marker', cursive;
}
.font-logo:hover {
    color: #00b894 !important;
}
```

În footer, se utilizează aceeași clasă pentru afișarea logo-ului. Codul existent în footer este:

```
<footer class="border-top footer text-muted">
    <div class="container">
        &copy; 2025 - ArticlesApp - <a asp-area="" asp-
controller="Home" asp-action="Privacy">Privacy</a>
    </div>
</footer>
```

Secvența de cod se modifică, folosind clasa *font-logo* pentru logo. De asemenea, se afișează și anul, folosind metoda *DateTime.Now.Year*. Codul devine:

```
<footer class="border-top footer text-muted">
    <div class="container">
        <p>&copy; @DateTime.Now.Year - <span class="font-
logo">ArticlesApp</span></p>
    </div>
</footer>
```

8. **Footerul** – să se stilizeze footer-ul, astfel încât să rămână fix în partea de jos a paginii.

```
.footer-position {
    position: fixed;
    left: 0;
    bottom: 0;
    width: 100%;
    background-color: #ffffff;
    text-align: center;
    box-shadow: 0 3px 12px #636e72;
}
```

9. **Culorile folosite** – pentru alegerea culorilor se poate utiliza <https://flatuicolors.com/>

10. **Pagina principală** – pe pagina principală **/Home/Index** să se afișeze pentru **utilizatorii neautentificați** doar 3 articole, în funcție de data publicării. Pentru a verifica dacă utilizatorul este autentificat, se folosește în metoda Index condiția **User.Identity.IsAuthenticated**.

Dacă este autentificat, se va face redirect către metoda Index din Controller-ul Articles utilizând **RedirectToAction("Index", "Article")**.

Dacă utilizatorul nu este autentificat, se preia primul articol folosind metoda **.First()**, iar pentru a prelua următoarele două articole se folosește **Skip(1).Take(2)** deoarece nu trebuie să afișăm încă o dată primul articol.

Primul articol să se afișeze folosind structura de mai jos. Să se utilizeze pentru titlu și conținut fonturi diferite. De exemplu, pentru titlu fontul Roboto, iar pentru conținut fontul Montserrat. Primul articol se va afișa pe un singur rând și pe o singură coloană, iar următoarele două articole pe un singur rând, dar pe două coloane. Pentru fiecare articol să se afișeze un link “Vezi articolul”.

De asemenea, pentru articolele de pe pagina principală să nu se afișeze tot conținutul. Să se afișeze doar câteva rânduri din conținutul fiecărui articol (se poate utiliza o clasă – căutați și studiați proprietatea line-clamp).

Apăsarea logo-ului trebuie să facă redirect către HomeController → Index.

Ruta default trebuie să fie configurată pentru a face redirect către HomeController → Index.

11. **Meniul** (Views -> Shared -> _Layout.cshtml) va afișa link-urile în funcție de rolul pe care îl are utilizatorul:

- **Utilizatorii neautentificați** nu o să vadă decât logo-ul aplicației (la apăsarea acestuia, utilizatorii o să fie redirecționați către pagina Index din HomeController)
- **Utilizatorul cu rolul Admin** poate accesa “Afișare articole”, “Afișare categorii”, “Adaugare articol”, “Afișare utilizatori”, “Colectii”
- **Utilizatorii cu rolul Editor** pot accesa “Afișare articole”, “Adaugare articol”, “Colectii”
- **Utilizatorii cu rolul User** pot accesa “Afișare articole” și “Colectii”

Implementare:

HomeController → metoda Index

```
public IActionResult Index()
{
    if (User.Identity.IsAuthenticated)
    {
        return RedirectToAction("Index", "Articles");
    }

    var articles = from article in db.Articles
                   select article;

    ViewBag.FirstArticle = articles.First();
    ViewBag.Articles = articles.OrderBy(o =>
o.Date).Skip(1).Take(2);

    return View();
}
```

Views → Home → Index.cshtml

```
<div class="row">
  <div class="col-12">
    <div class="bg-light rounded-3 p-5 mb-4">
      <div class="row">
        <div class="col-md-8">
          <h1 class="font-roboto display-5 fw-
bold">@Html.Raw(ViewBag.FirstArticle.Title)</h1>
          <div class="font-montserrat mb-3
truncate">@Html.Raw(ViewBag.FirstArticle.Content)</div>
          <a href="/Articles/Show/@ViewBag.FirstArticle.Id"
class="btn btn-outline-success">Vezi articolul</a>
        </div>
        <div class="col-md-4">
          
        </div>
      </div>
    </div>
  </div>
</div>

<div class="row mb-4">
  @foreach (var article in ViewBag.Articles)
  {
    <div class="col-6 row-style">
      <div class="card">
        <div class="card-title p-3 text-center bg-dark text-
white">
          <h4>@Html.Raw(article.Title)</h4>
        </div>
        <div class="card-body">
          <div class="font-montserrat
truncate">@Html.Raw(article.Content)</div>
        </div>
        <div class="card-footer">
          <a class="btn btn-outline-success"
href="/Articles/Show/@article.Id">Vezi Articolul</a>
        </div>
      </div>
    </div>
  }
</div>
```