ДИПЛОМЕН ПРОЕКТ ЗА ДЪРЖАВЕН ЗРЕЛОСТЕН ИЗПИТ

по професия код 481030 „Приложен програмист“

специалност код 4810301 Приложно програмиране“

Тема: „Сайт за новинарски публикации“

Автор:

Васил Тодоров Станчев, клас XII В

Ръководител:

Антони Димитров

Бургас

СЪДЪРЖАНИЕ

[1 Увод 3](#_Toc129695703)

[2 Цели и обхват на софтуерното приложение 3](#_Toc129695704)

[3 Анализ на решението 4](#_Toc129695705)

[3.1 Потребителски изисквания и работен процес 4](#_Toc129695706)

[3.2 Примерен потребителски интерфейс 5](#_Toc129695707)

[3.3 Диаграми на анализа 5](#_Toc129695708)

[3.4 Модел на съдържанието / данните 5](#_Toc129695709)

[4 Дизайн 6](#_Toc129695710)

[4.1 Реализация на архитектурата на приложението 6](#_Toc129695711)

[4.2 Описание на слоевете, предназначението им, библиотеки и методи включени в съответния слой. 6](#_Toc129695712)

[4.3 Организация и код на заявките към база от данни 6](#_Toc129695713)

[4.4 Наличие на потребителски интерфейс (конзолен, графичен, уеб) 6](#_Toc129695714)

[5 Ефективност и бързодействие на решението 6](#_Toc129695715)

[6 Тестване 7](#_Toc129695716)

[7 Заключение и възможно бъдещо развитие 7](#_Toc129695717)

[8 Използвани литературни източници и Уеб сайтове 7](#_Toc129695718)

[9 Приложения 7](#_Toc129695719)

[10 Критерии и показатели за оценяване 9](#_Toc129695720)

# Увод

Настоящият документ представлява описание на дипломен проект за Държавен зрелостен изпит. Проектът представлява сайт за новинарски публикации. В текущия документ е описано подробно цялата същност на идеята зад проекта, анализ на самото решение, диаграми, дизайн на страницата, тестване и възможно бъдещо развитие.

# Цели и обхват на софтуерното приложение

## Цели

Целта на проекта е направата на сайт за новинарски публикации. Все по често се наблюдава липсата на стойностни сайтове за свободна журналистическа практика. С този сайт се цели промяната на това. Сайтът разполага с регистрация, която цели да разграничава по-заинтересованите потребители, като всички публикации биват разделени на одобрени и неодобрени. Тези публикации могат да бъдат създавани, променяни или изтривани от потребители с ролята редактор, предварително вкарани от администратора на сайта. След това тези публикации биват одобрени и те стават видими за нерегистрирани потребители. Ролята администратор има права над всичко свързано с сайта, включително и разпределянето на потребители предварително одобрени като редактори.

## Обхват

Обхватът на проекта е към абсолютно всички. Сайтът цели да направи медиите една по приятна среда за всички хора като това се цели чрез свободата на публикациите. Всеки е приветстван да се регистрира и да разглежда всички публикации. Сайтът също цели да подтиква млади или опитни журналисти да развиват своя опит и гледна точка, като им дава възможността да изразят важни за тях теми.

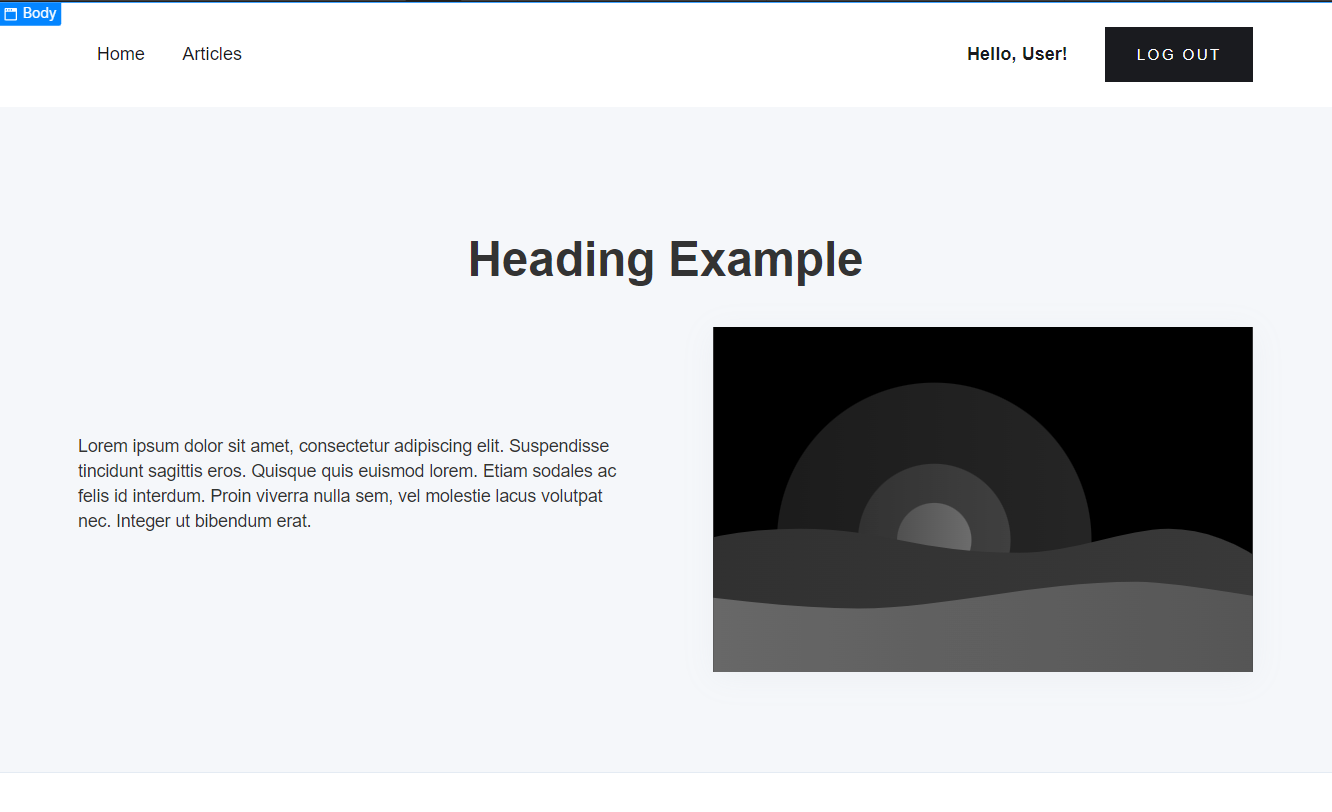
# Анализ на решението

## Потребителски изисквания и работен процес

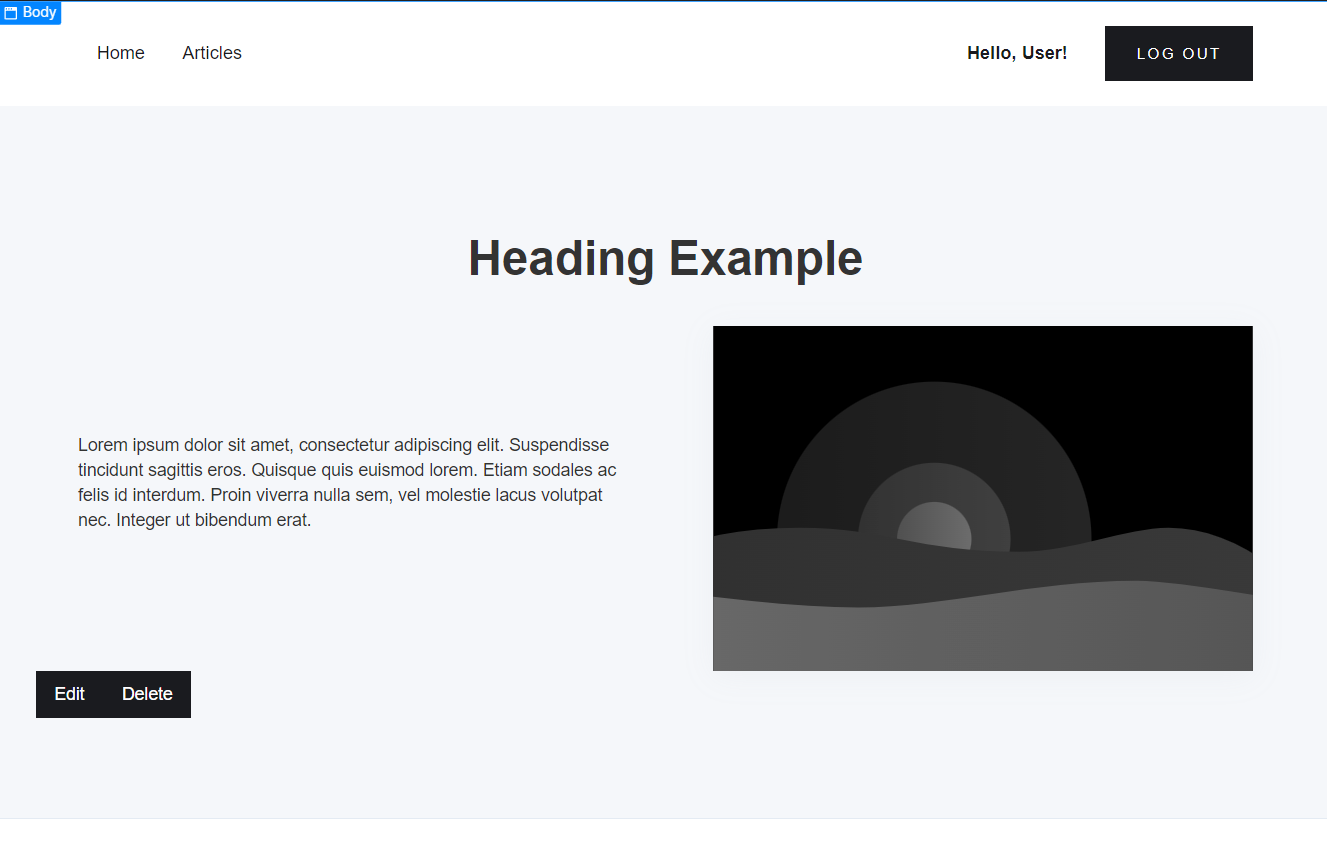
Всички потребители в сайта биват разпределяни в четири групи, всеки с различни права и нива на достъп на редакция и видимост. Входът за всеки потребител е един, като тяхната информация бива съхранявана в база данни. Всеки потребител разполага с профил, като за входни данни се приемат неговия имейл и парола. С долу описаните диаграми може да бъде придобито по-голяма представа за ролите, примерен потребителски интерфейс и др.

## Примерен потребителски интерфейс

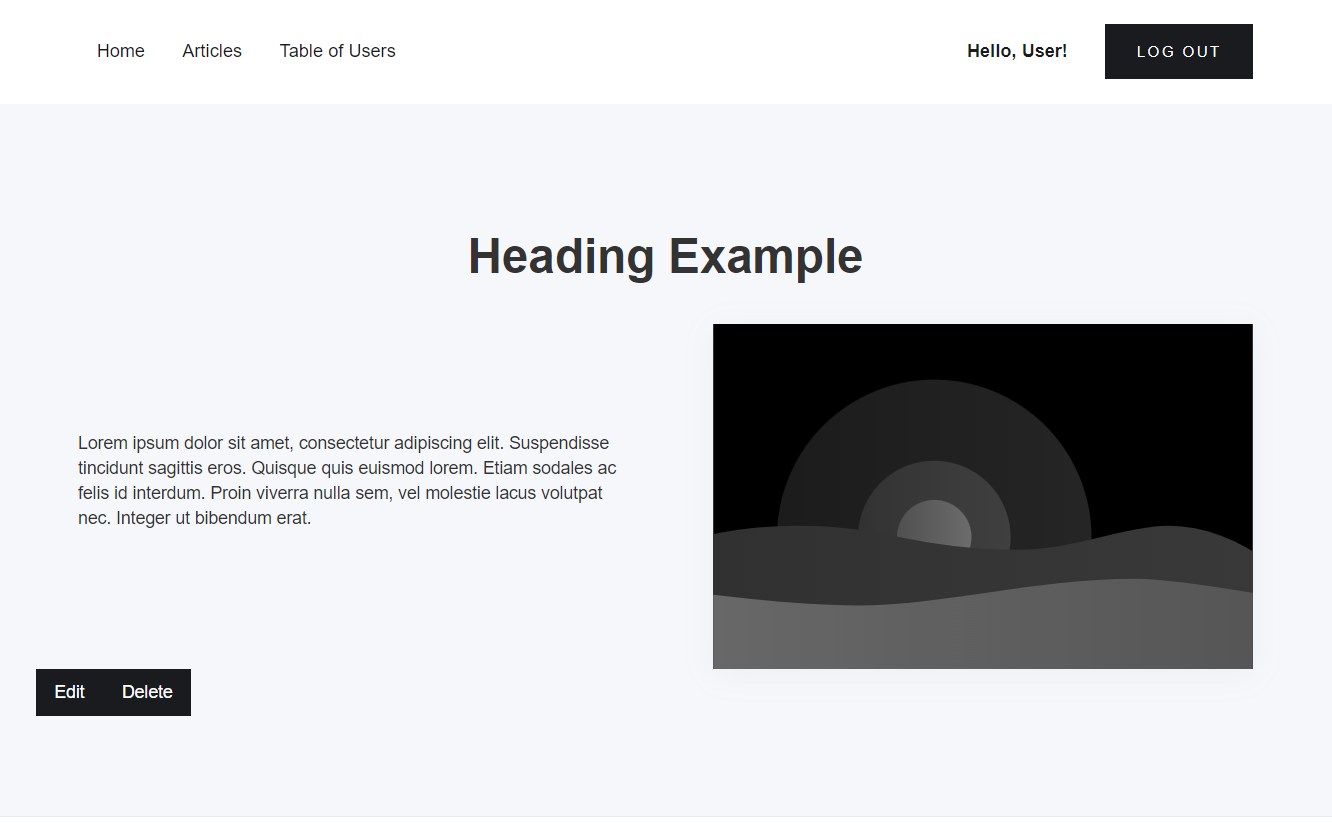
Фигурите 1, 2 и 3 демонстрират примерни потребителски интерфейси за различните групи от потребители. Те са представени в следната последователност: регистриран потребител, редактор и администратор.



Фигура 1



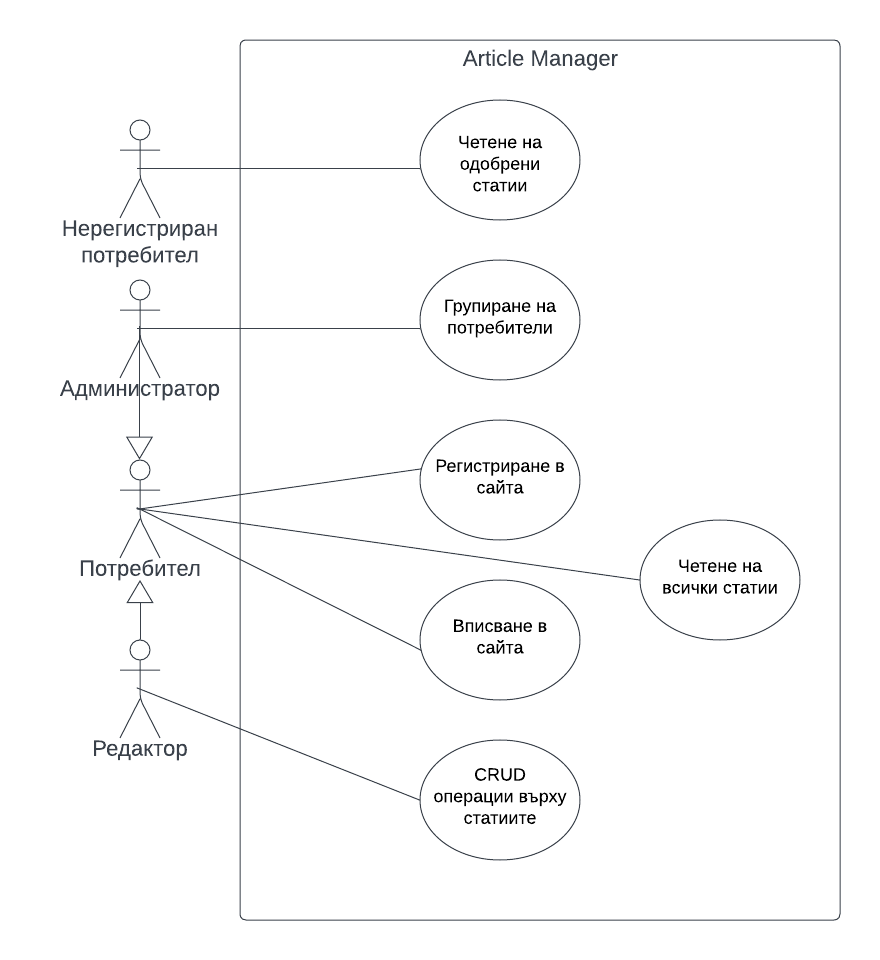
Фигура 2



Фигура 3

## Диаграми на анализа

### Диаграма на използване на казус

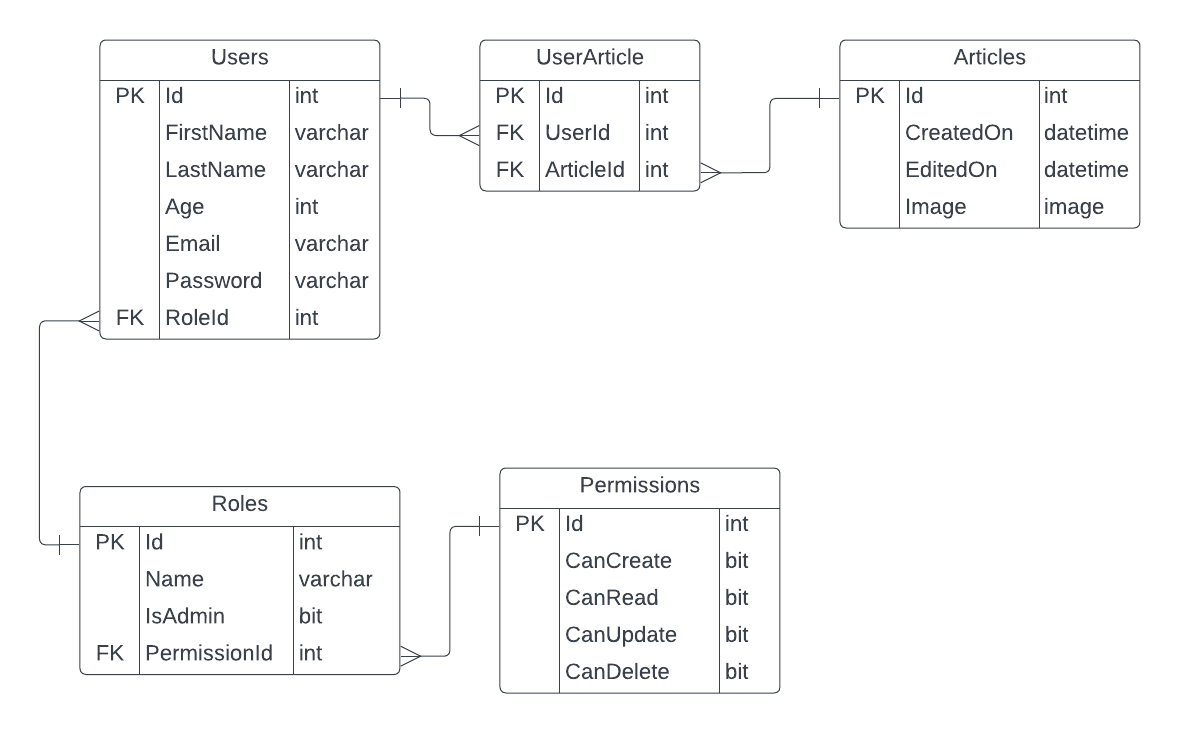


Фигура 4

Фигура 4 демонстрира основните действия на различните групи от потребители. Те са четири:

* Нерегистриран потребител – това е най-простият тип потребител, който без регистрация може да види вече одобрените публикации, като ако желае да види останалите, та той става обикновен потребител.
* Потребител – това са потребители, които желаят да видят по-голяма част от сайта и следователно могат да си направят регистрация или да се впишат ако вече имат. С това те вече имат достъп до абсолютно всички публикувани статии.
* Редактор – това са потребители, предварително одобрени от администратор, които имат права над създаването, редактирането и изтриването на публикации. Редактора разполага също със същите права като обикновен потребител.
* Администратор – това е потребител, който е с най-много правомощия над сайта. Той има права като редактор, като допълнение той е отговорен за групирането на потребителите.

### E/R диаграма

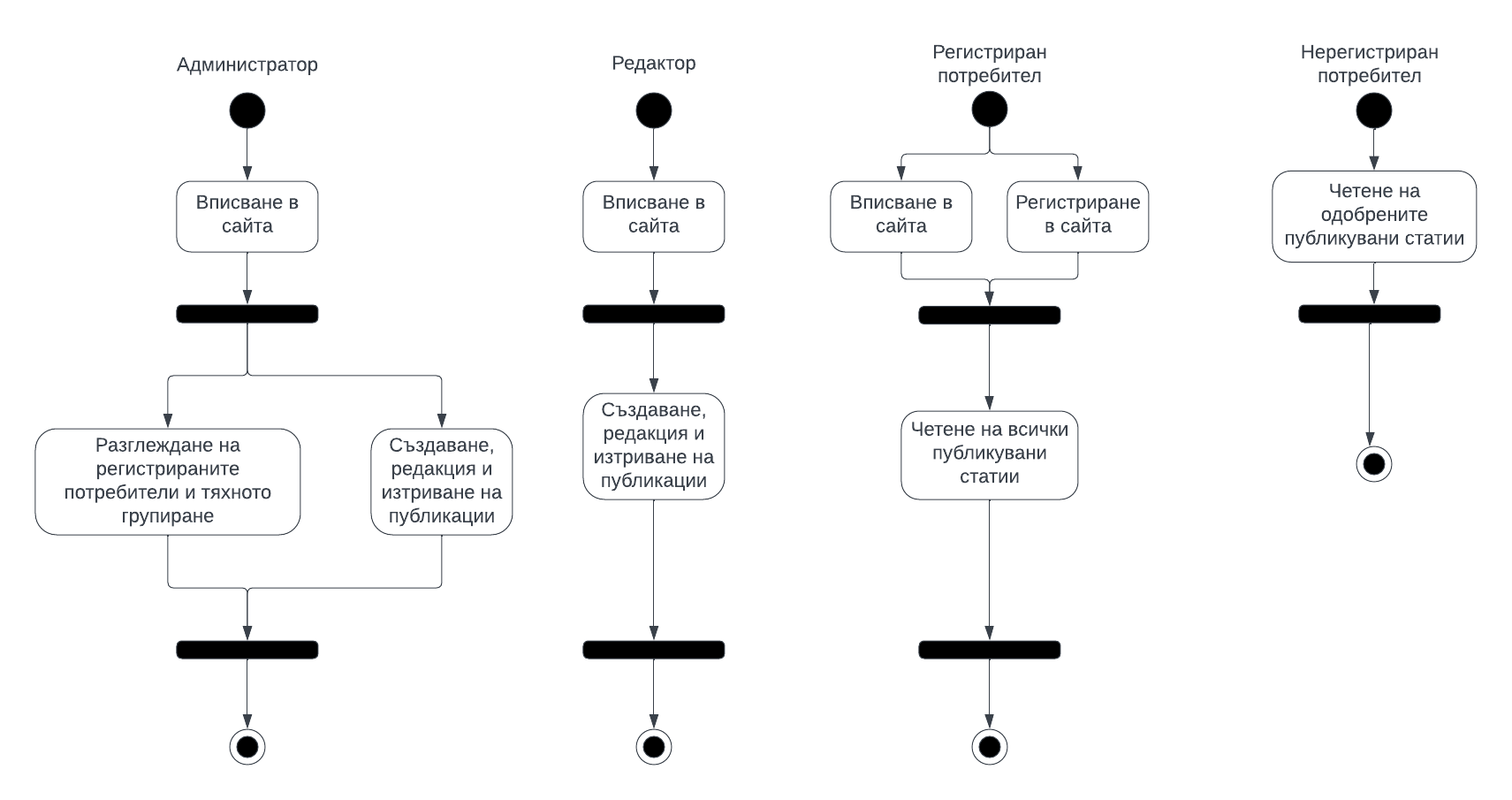


Фигура 5

Фигура 5 демонстрира структурата на базата данни използвана за сайта. Имаме 4 основни таблици и една междинна за връзката между редактор и статията, върху която е работил.

Таблицата свързана със статиите съдържа информация относно датите на промяна и създаване на самата публикация, както и поле за снимка на статията. Таблицата за потребителите съдържа основна информация за всеки от профилите на потребителите като име, години, имейли, пароли и роли. Таблицата за роли съдържа информация за 4-те роли, в които са групирани потребителите. Към тази таблица има помощна таблица, която определя различните права на самите роли.

### Диаграма на активност

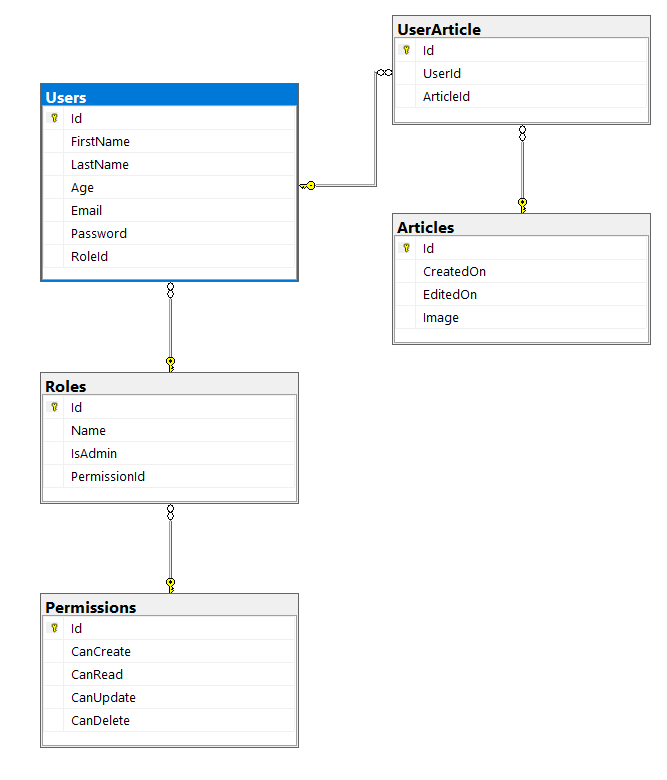


Фигура 6

Фигура 6 демонстрира основните възможни действия на всяка една от четирите групи. Администратора след вписване има възможността да наблюдава в табличен вид всички регистрирани потребители и може да ги разпределя в другите роли, да промени обикновения потребител в редактор. Той също има права над публикациите. Редактора след вписване има права над публикациите. Той може да създава, редактира и изтрива. Регистрирания потребител може да се впише в своя профил или да си направи такъв с регистрация. След което може да разгледа всички създадени публикации. Нерегистрираните потребител може да разгледа одобрените публикации.

## Модел на съдържанието / данните

Данните в сайта биват съхранявани в база данни реализирана с технологията MSSQL. Фигура 7 демонстрира това. Базата данни е нормализирана в трета норма форма.



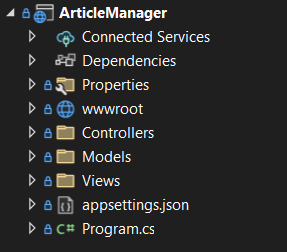
Фигура 7

# Дизайн

Проектът е разработен на платформата “ASP.NET Core” с езика “C#”. Проекта е базиран на структурата MVC(Модел-изглед-контролер). Структурата MVC е популярен шаблон за проектиране, използван в разработката на софтуер, включително в “ASP.NET Core”. Използва се за разделяне на проблемите на дадено приложение на три основни компонента: модел, изглед и контролер.

## Реализация на архитектурата на приложението

### Архитектура на проекта



Фигура 8

Фигура 8 изобразява архитектурата на проекта. Всеки един от различните слоеве е разделен в различна папка.

## Описание на слоевете, предназначението им, библиотеки и методи включени в съответния слой.

### Модели

Компонентът “Модел” представлява данните и бизнес логиката на приложението. Обикновено включва класове и структури от данни, които взаимодействат с източниците на данни на приложението, като бази данни. В проектът този компонент на модела е реализиран с помощта на класове и технологии за достъп до данни с помощта на “Entity Framework”.

### Изгледи

Компонентът “Изглед” представлява потребителския интерфейс на приложението. Той определя как данните на приложението се представят на потребителя и как потребителят взаимодейства с тях. В проектът този компонент е реализиран чрез синтаксиса на “Razor” за генериране на “HTML”, “CSS” и “JavaScript” код.

### Контролери

Компонентът “Контролер” действа като посредник между компонентите “Модел” и “Изглед”. Той обработва потребителски заявки, извлича данни от модела и предава тези данни на изгледа за изобразяване. В проектът компонентът “Контролер” е реализиран с помощта на класове, които обработват входящи “HTTP” заявки и ги свързват с подходящи действия.

## Организация и код на заявките към база от данни

Проектът е реализиран с методиката “Database First Approach”. По тази методика, базата данни е създадена първа и след това таблиците биват реализирани в компонента “Модел”.

За реализацията на връзката между базата и проекта са използвани методи от “Nuget Packets” на “Visual Studio” и по-точно “Microsoft.EntityFrameworkCore.SqlServer”, който позволява това. Самата заявка бива съхранявана в стрингов низ, който след това бива използван в метода за връзка.

## Наличие на потребителски интерфейс (конзолен, графичен, уеб)

Потребителският интерфейс представлява уеб страници. Те са създадени като компоненти в “Изгледи”. Те биват създадени с помощта на “Razor” методиката, която е специализирана за технологията “ASP.NET Core”. В точка 3.2 е представен примерен потребителски интерфейс с който може да бъде придобита представа за структурата на самата страница.

# Ефективност и бързодействие на решението

Настоящ проект не е първия по рода си. Налице са много новинарски сайтове, но настоящият проект се отличава с черти, които могат да бъдат счетени за коренно променящи. Една от тях е различното ниво на видимост на самите публикации. С разделянето им на одобрени и неодобрени, може да бъде предизвикан по-голям интерес у потребителите, които желаят да са по-запознати с всички подробности, колкото и малки, в днешната журналистика.

# Тестване

За тестването на ефективността на сайта беше използвана методът “NUnit”. В тази точка следват описание на тази методика и няколко примерни тестови случая.

## Какво представлява NUnit тестване

“NUnit” е един от най-популярните инструменти за тестване на софтуер в “.NET” екосистемата. Той предоставя библиотека за управление на тестовете, както и инструмент за изпълнение на тези тестове. Тестовете са написани в специален синтаксис, който позволява на “NUnit” да идентифицира тестовете и да ги изпълнява автоматично.

Тестовете в “NUnit” могат да бъдат организирани в няколко класа, като всеки клас представлява един тестов сценарий. Всяка тестова функция е отбелязана с атрибут, който указва, че това е тестова функция и какъв е очакваният резултат.

## Примерни тестови случаи

# Заключение и възможно бъдещо развитие

## Заключение

Проектът за новинарски публикации беше реализиран успешно. Проектирано и разработено беше функционално решение за такъв сайт, който позволява на потребителите да четат новинарски публикации, които напълно се вписват в идеята за свободна и по-достъпна журналистика. Сайтът се възползва от възможностите на технологията “ASP.NET Core” и езикът “C#”, за да създаде ефективен и лесен за използване уеб сайт, който отговаря на всички изискванията и очакванията. Проектът е завършен с цел, че ще служи като ценен ресурс за всички хора от обществото, като предоставя достъпна и лесна информация.

## Бъдещо развитие

Мислейки за бъдещо развитие, проектът може да бъде доразвит към няколко области, в които ще може допълнително да се подобри функционалността и потребителското изживяване. Една област на подобрение може да бъде прилагането на система за препоръки, която предлага статии на потребителите въз основа на тяхната история на четене или предпочитания. Това ще подпомогне с персонализирането на потребителското изживяване и ще ги задържи ангажирани със сайта за по-дълго време.

Друга област на бъдещо развитие може да бъде въвеждането на мултимедиен компонент в новинарския сайт, като видео или подкаст. Това ще разшири вида съдържание, достъпно за потребителите, и ще направи сайта по-ангажиращ и динамичен.

# Използвани литературни източници и Уеб сайтове

* (ASP.NET Core, n.d.)
* (Lock, 2021)
* (Flagman, n.d.)

# Приложения

# Критерии и показатели за оценяване

|  |  |  |
| --- | --- | --- |
| Критерии и показатели за оценяване | Максимален брой точки за показателите | Максимален брой точки за критерия |
| 1. Съответствие с изискванията за съдържание и структура на дипломния проект |  | 20 |
| 1. 1. логическа последователност и структура на изложението, балансиране на отделните части | 4 |  |
| 1.2. задълбоченост и пълнота при формулиране на обекта, предмета, целта и задачите в разработването на темата | 7 |  |
| 1.3. използване на подходящи изследователски методи | 4 |  |
| 1.4. стил и оформяне на дипломната работа (терминология, стил на писане, текстообработка и оформяне на фигури и таблици) | 5 |  |
| 2. Съответствие между поставените цели на дипломния проект и получените резултати |  | 20 |
| 2.1. изводите следват пряко от изложението, формулирани са ясно, решават поставените в началото на изследването цели и задачи и водят до убедителна защита на поставената теза | 10 |  |
| 2.2. оригиналност, значимост и актуалност на темата | 6 |  |
| 2.3. задълбоченост и обоснованост на предложенията и насоките | 4 |  |
| 3. Представяне на дипломния проект |  | 20 |
| 3.1. представянето на разработката по темата e ясно и точно | 5 |  |
| 3.2. онагледяване на експозето с:  а) презентация;  б) графични материали;  в) практически резултати;  г) компютърна мултимедийна симулация и анимация | 10 |  |
| 3.3. умения за презентиране | 5 |  |
| 4. Отговори на зададените въпроси от рецензента и/или членовете на комисията за защита на дипломен проект |  | 30 |
| 4.1. разбира същността на зададените въпроси и отговаря пълно, точно и убедително | 10 |  |
| 4.2. логически построени и точни отговори на зададените въпроси | 10 |  |
| 4.3. съдържателни и обосновани отговори на въпросите | 10 |  |
| 5. Използване на професионалната терминология, добър и ясен стил, обща езикова грамотност |  | 10 |
| 5.1. Правилно използване на професионалната терминология | 5 |  |
| 5.2. Ясен изказ и обща езикова грамотност | 5 |  |
| Общ брой точки: | Максимален бр. точки 100 | Максимален бр. точки 100 |