

ДИГИТАЛНА ДОСТЪПНОСТ

ТЕХНИЧЕСКА ДОКУМЕНТАЦИЯ

Разработил: Ангел Лъчезаров Чолаков
имейл: angel.lachezarov@gmail.com

СЪДЪРЖАНИЕ

БЛАГОДАРНОСТИ.....	3
ЦЕЛИ И ПРИЧИНИ ЗА РАЗРАБОТКА	4
СТРУКТУРА НА ПРОЕКТ	5
СТРУКТУРА НА GITHUB	5
ОБЩ ПЛАН НА КОМУНИКАЦИЯ МЕЖДУ ПРИЛОЖЕНИЯТА.....	6
ЛОГИЧЕСКА СХЕМА НА БАЗАТА ДАННИ	7
СХЕМА НА БАЗА ДАННИ	9
КОНТРОЛЕН ПАНЕЛ.....	10
ПОТРЕБИТЕЛИ И РОЛИ	11
ПОТРЕБИТЕЛИ	12
АРХИТЕКТУРА НА ПРОЕКТ	12
КОМУНИКАТОР	14
АРХИТЕКТУРА НА ПРОЕКТ	14
„ДИГИТАЛНА ДОСТЪПНОСТ“.....	15

ПОСВЕЩЕНИЕ

Посвещавам настоящия проект в памет на вуйчо ми Георги за това, че успя да ме вдъхнови за идеята и концепцията на този проект.

БЛАГОДАРНОСТИ

Благодаря на екипа, разработил софтуерното приложение “NeuralSpeechLab” за предоставянето на лиценз за програмата им за гласов синтез, както и за предоставеното съдействие при конфигурацията на техния продукт. Той е изключително важна част от гласовия компонент на разгледания в този документ софтуер.

ЗА СОФТУНИ

Това е проект за курса “Asp.Net Advanced”. Курсовият проект представлява интернет приложение, чрез което се управляват и администрират потребителите на „Дигитална достъпност“ – настолно приложение за управление на компютър от незрящи лица. За оценяването да се зачитат контролната система (ASP.NET MVC проекта) и комуникатора с настолната програма (ASP.NET API проекта).

ЦЕЛИ И ПРИЧИНИ ЗА РАЗРАБОТКА

„Дигитална достъпност“ е настолно приложение, чрез което незрящи лица могат да използват компютър за изпълнение на основните му функционалности по възможно най-лесния за тях начин. Главната цел на платформата е да улесни и ускори процеса на използване на компютър, чрез въвеждане на универсален метод за управление, базиран на телефонния оператор, гласово въвеждане на текст и разбираеми гласови съобщения. Целевата група на проекта са незрящи лица с ниска компютърна компетентност, възрастни лица.

Главният мотив за разработката на „Дигитална достъпност“ е желанието ми да направя алтернатива на сложните за научаване софтуери тип „Екранен четец“, която да предлага по-лесен, универсален и бърз метод за управление на компютър.

Приложението „Дигитална достъпност“ е основан на принцип за управление, подобен на този от телефонните оператори. Чрез това приложение, незрящото лице управлява компютъра чрез натискане на съответния на функцията цифров бутон. Потребителят разбира кое число на коя функция отговаря чрез гласови инструкции, които компютърът „диктува“ на потребителят всеки път при избиране на съответната функция.

Контролният панел на „Дигитална достъпност“ е уеб-базирано приложение, което обединява процесите на управление на настолната програма. Чрез този панел администраторите на незрящи потребители могат да издават лицензи за „Диг. Достъпност“, да управляват техните настройки и предпочитания, както и да задават въпроси. За да извършват изброените операции, те трябва да бъдат предварително одобрени от избрана от тях организация за работа с незрящи, слепи или слабовиждащи (напр. Съюз на слепите в България)*. Служителите на организацията отговарят на зададените въпроси и се грижат за правилния подбор на администратори.

Мотивът за разработката на контролния панел е желанието ми да улесня процеса на управление на настолното приложение, като предоставя възможност за отдалечено администриране – достъпно от всяко устройство, от всяка точка на света.

* Предварителното одобряване е реализирано с цел защита на личните данни на незрящите.

СТРУКТУРА НА ПРОЕКТ

Проектът се състои от три отделни софтуерни приложения, написани на програмния език C#:

1. **„Дигитална достъпност“** – представлява настолно приложение. Предназначено е за незрящите потребители, които чрез него упражняват основната цел на разработката. Това приложение е от тип Windows Forms.
2. **Контролен панел** – представлява уеб-базирано приложение. Предназначено е за администраторите на незрящите потребители, служителите и директорите на организациите/фондациите за незрящи. Посредством това приложение се управляват лицензите, настройките и функционалностите на потребителите на настолното приложение (повече в секцията „Потребители и роли“), както и се задават/отговарят въпросите на потребителите. Това приложение е от тип ASP.NET MVC.
3. **Комуникатор** – представлява уеб-базиран интерфейс за комуникация с външни системи (API), посредством който настолното приложение (т.1) комуникира с базата данни. Този проект е от тип ASP.NET API

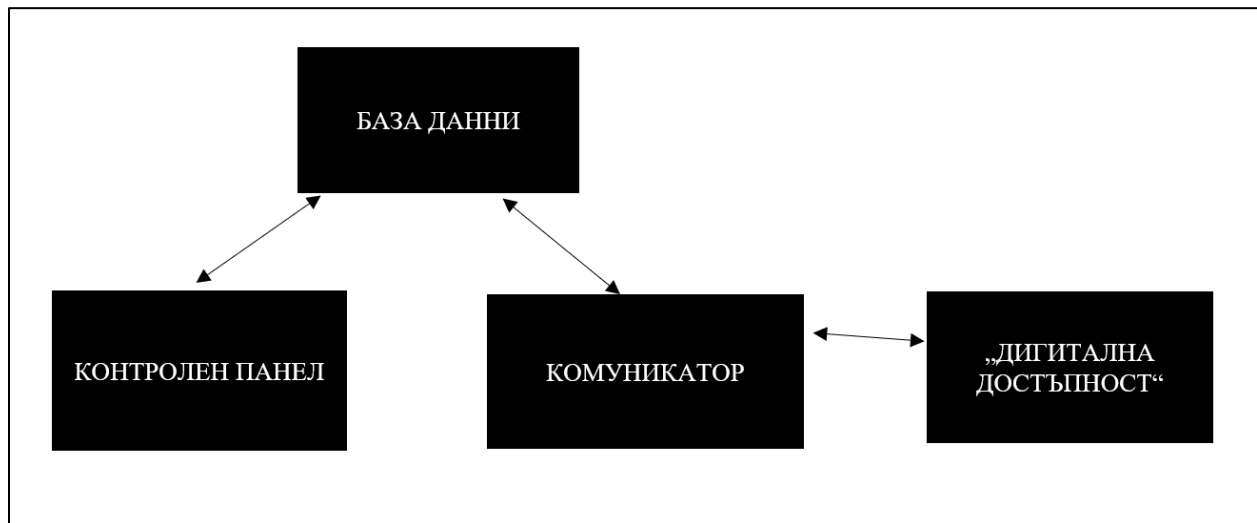
Приложението от т.1 е изградено с .NET Framework 4.6, а тези от т.2 и т.3 са изградени с .NET 8.

За СофтУни: За част от проекта за курса “Asp.Net Advanced” да се считат т.2 и т.3.

СТРУКТУРА НА GITHUB

На главната страница на проекта в GitHub (“<https://github.com/cholakov-angel/DigitalAccessibility-ControlPanel>”) се намират проектните файлове на контролния проект. API проектът се намира в папка (“DigitalAccessibility/API”), а настолното приложение се намира в папка (“DigitalAccessibility/Forms”).

ОБЩ ПЛАН НА КОМУНИКАЦИЯ МЕЖДУ ПРИЛОЖЕНИЯТА

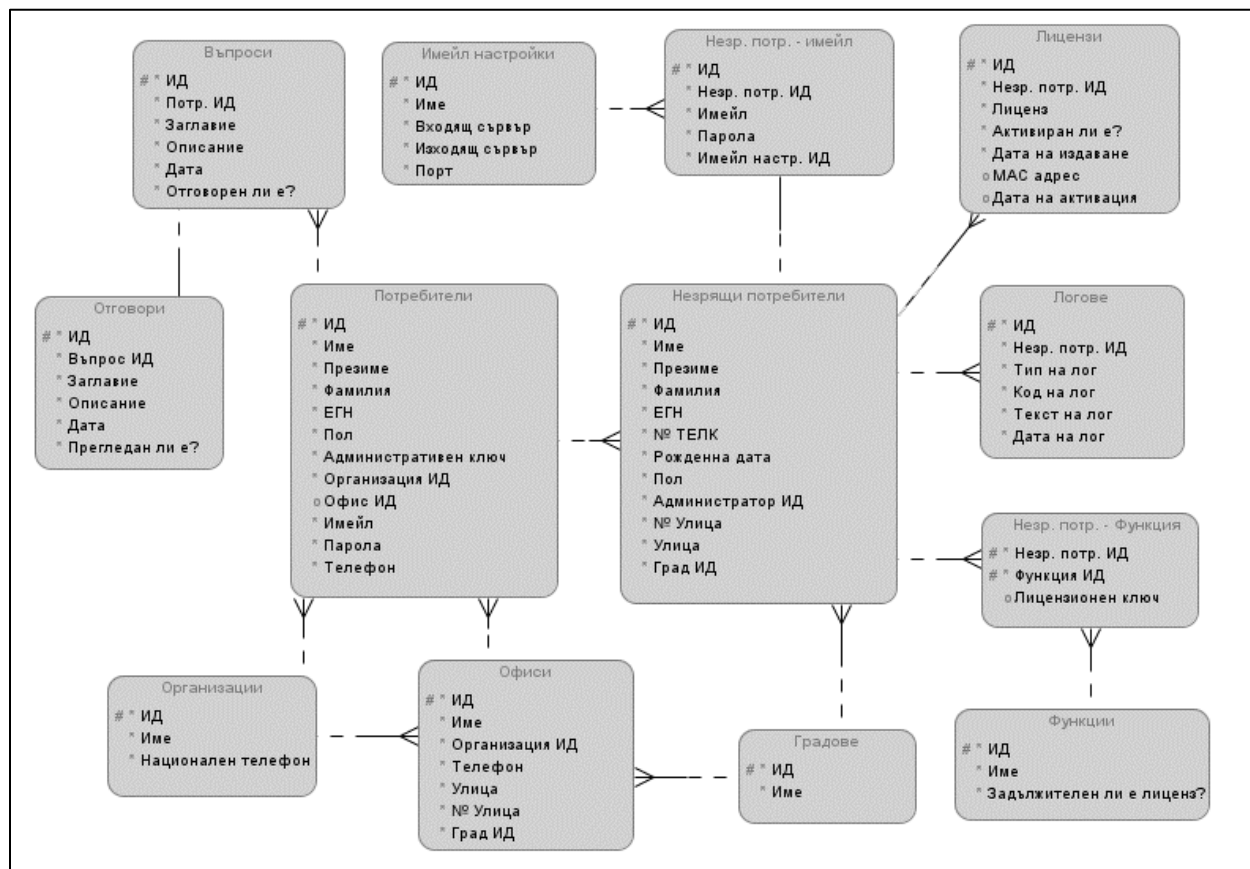


Контролният панел (MVC проекта), посредством „услуги“ (services) се извършват четене и манипулация базата данни – добавяне, изтриване и редактиране на записи.

Комуникаторът (API проекта), посредством „услуги“ се извършва четене и манипулация на базата данни – добавяне на записи. Редактиране и изтриване не се извършват.

„Дигитална достъпност“ (WinForms проекта) не комуникира пряко с базата данни. За тази цел, приложението се свързва чрез HTTP заявки към комуникатора, който извършва операциите.

ЛОГИЧЕСКА СХЕМА НА БАЗАТА ДАННИ



Обяснение на функцията на таблиците:

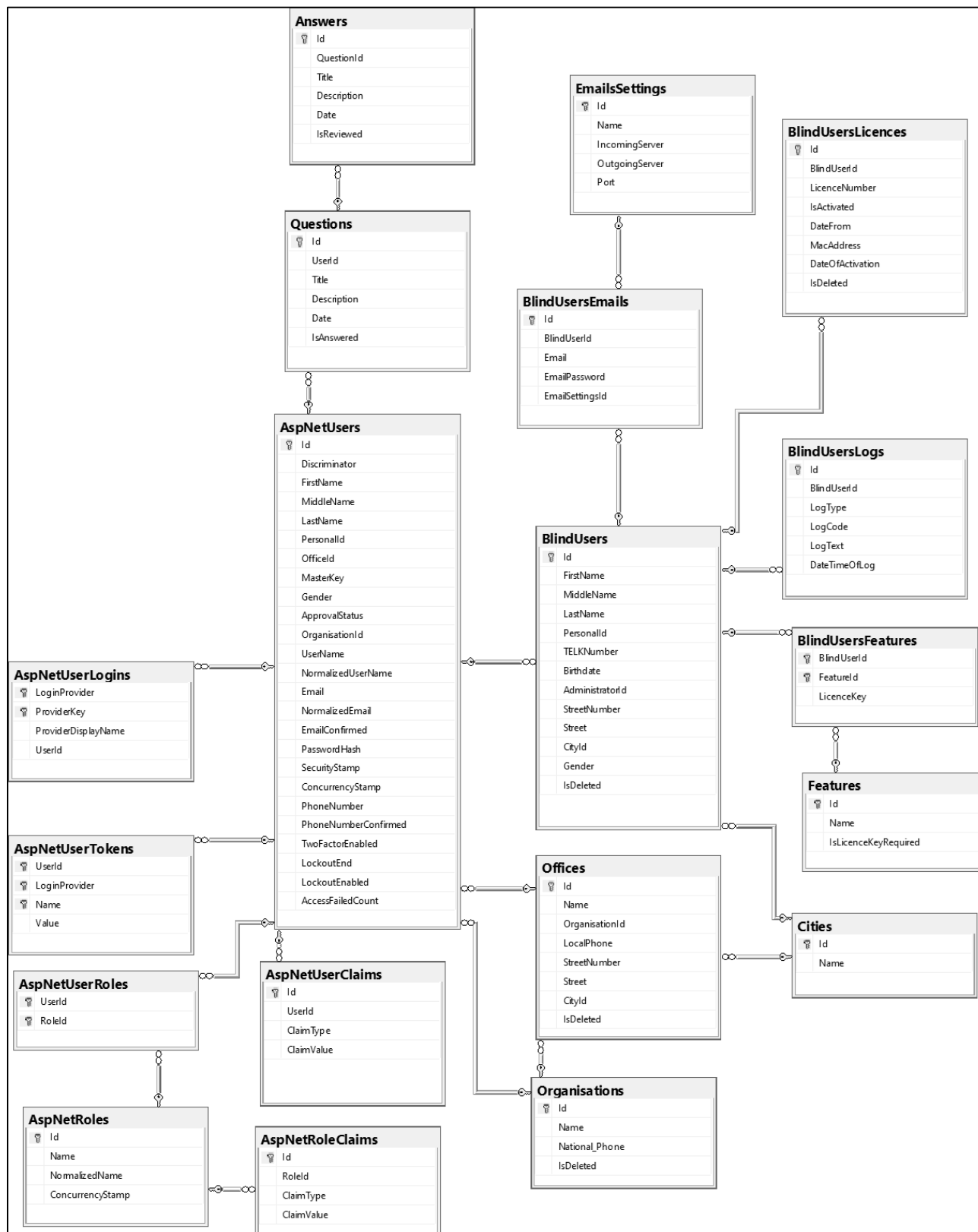
1. „Потребители“ – съдържа информацията за потребителите на контролния панел;
2. „Незрящи потребители“ – съдържа информация за незрящите потребители на „Дигитална достъпност“;
3. „Имейл настройки“ – съдържа информация за входящ/изходящ имейл сървър и порт;
4. „Имейли“ – съдържа информация за имейл и парола на незрящите потребители;
5. „Логове“ – съдържа информация за настъпилите съобщения и грешки по време на работата на незрящите потребители с настолното приложение;
6. „Функционалности“ – съдържа информация относно предлаганите от „Дигитална достъпност“ допълнителни функционалности;

7. „Незр. Потр - Функция.“ – съдържа информация относно активираните функционалности за незрящите потребители;
8. „Въпроси“ – съдържа информация за зададените от администраторите на незрящите към служителите на организацията;
9. „Отговори“ – съдържа информация за отговорите на въпросите;
10. „Организации“ – съдържа информация относно организациите/фондациите, занимаващи се с незрящи лица;
11. „Офиси“ – съдържа информация относно клоновата мрежа на организацията;
12. „Градове“ – съдържа информация относно областните градове в България;
13. „Лицензи“ – съдържа информация за лицензите на незрящите потребители за достъп до настолното приложение.

Разяснения относно логическите особености на приложението, свързани с базата данни:

1. Един незрящ потребител има право до 3 лиценза за „Дигитална достъпност“;
2. Дата на раждане и пол на потребителите и незрящите лица се изчислява автоматично от въведеното ЕГН;
3. Един незрящ потребител има право на един имейл адрес;
4. Таблицата „ЛОГОВЕ“ се попълва единствено от настолното приложение (посредством API приложението).
5. Таблицата „Потребители“ е реализирана като допълнение на стандартната за Asp.Net таблица “Asp.NetUsers”

СХЕМА НА БАЗА ДАННИ



Уточнения:

1. В полето с име "PersonalID" в таблиците „AspNetUsers" и "BlindUsers" се съхранява ЕГН.
2. В полето с име "TELKNumber" в таблицата "BlindUsers" се съхранява номера на ТЕЛК на незрящите лица.
3. В полето с име "MasterKey" в таблицата "AspNetUsers" се съхранява административния ключ за потребителите от роля "UserAdministrator" (повече в секция „Потребители и роли")

Предварително попълване на данни:

В миграцията, намираща се в с MVC проекта, са предвидени данни за базата данни, посредством които да се улесни проверяването на проекта. Попълнените данни са релевантни за типа на информацията, но са недействителни. Поради естеството на съхраняваните данни, таблица "BlindUsersEmails" не е предварително попълнена.

КОНТРОЛЕН ПАНЕЛ

Контролният панел представлява уеб-базирана платформа, чиято цел е да улесни следните процеси:

1. Проверка и одобряване на лицата, желаещи да станат администратори на незрящи лица;
2. Управление на лицензирането, настройките, предпочитанията и функционалностите на незрящите лица от страна на администраторите;
3. Добавяне на организациите и фондациите, занимаващи се с незрящи лица;
4. Управление на служителите на организациите, които имат достъп до системата.

ПОТРЕБИТЕЛИ И РОЛИ

Контролният панел е предназначен за използване от следните типове потребители:

1. **Администратор на незрящи** – отговаря за незрящите потребители на настолното приложение. Той ги добавя към системата, управлява техните настройки за функционалности, имейл настройки. Генерира необходимите за работата на „Дигитална достъпност“ активационни лицензи и преглежда съобщенията за информация и грешка (логове), настъпили при работата на поверените му незрящи лица. Този потребител предварително избира предпочитана организация и нейн клон, като може да задава въпроси към нея;
2. **Служител на клон на организация** – отговаря за определянето кои кандидати за потребител тип „администратор на незрящи“ да бъдат допуснати до системата. Може да преглежда информацията относно тях, да отговаря на зададените въпроси и при доказана злоупотреба да ги блокира;
3. **Управител на клон на организация** – отговаря за добавянето и премахването на подчинените, които ще имат достъп до системата. Редактира информацията за поверения му клон;
4. **Управител на организация** – отговаря за добавянето и премахването на клонове на организацията му. Може да променя информацията за всеки един клон, както и да добавя/сменя управителя. Редактира информацията за организацията;
5. **Администратор** – отговаря за добавянето и премахването на организации, както и техните управители.

За практическата реализация на потребителската система, са използвани следните роли и зони (“Areas”)

1. **“UserAdministrator”** – отговаря за функциите на администратора на незрящи;
2. **“OfficeWorker”** – отговаря за функциите на служителя на клон на организация;
3. **“OfficeAdministrator”** – отговаря за функциите на управителя на клон на организация;
4. **“OrgAdministrator”** – отговаря за функциите на управителя на организация
5. **“Admin”** – отговаря за функциите на администратора;

6. **“WaitingApproval”** – допълнителна роля и “area”, предназначена за потребителите в процес на изчакване на одобрение.

ПОТРЕБИТЕЛИ

За по-доброто тестване на приложението са създадени тестови потребители.

ИМЕЙЛ	ПАРОЛА	ЛОГИЧЕСКА РОЛЯ	РОЛЯ В ASP.NET
admin@admin.com	Admin1.Admin	Администратор	Admin
horizonti@org.com	Stefan1.Stefanov	Управител на организация	OrgAdministrator
angel.borisov@admin.com	Angel1.Borisov	Администратор на незрящи	UserAdministrator
zoia.stefanova@worker.com	Zoia1.Stefanova	Служител на клон на организация	OfficeWorker
nikolay.nikolov@admin.com	Nikolay1.Nikolov	Администратор на незрящи	UserAdministrator
blindunion@org.com	Petar1.Petrov	Управител на организация	OrgAdministrator
petar.petrov1@gmail.com	Petar1.Petrov1	Управител на клон на организация	OfficeAdministrator

АРХИТЕКТУРА НА ПРОЕКТ

За реализирането на контролния панел, са направени 9 проекта, 8 от които са от тип “ClassLibrary” и един главен от тип “Asp.Net MVC”. Разделени са по предназначение в папки по следния начин:

1. **“Common”** – съдържа общи статични константи и методи:
 - 1.1 “DigAccess.Common” – състои се от статични класове, в които се съхраняват константите (формат на дата, съобщения за неправилно въведени данни във формулярите), метод за преобразуване на символен низ в уникален идентификатор (“Guid”) и метод за извличане на рождена дата и пол от ЕГН.
2. **“Data”** – съдържа проекти, свързани с базата данни:
 - 2.1. “DigAccess.Data.Entities” – състои се от класовете, които се използват като модели (Entities) за базата данни;
 - 2.2. “DigAccess.DbContext” – състои се от основния за Entity Framework клас “DbContext”, който съдържа информация за таблиците, както и техните

свойства (външни ключове, ограничения, стойности по подразбиране).

Съхранява се и папката с миграции;

- 2.3. “DigAccess.Data.Seeder” – състои се от класовете, които се използват за автоматичното попълване на базата данни при нейното създаване.
- 3. “**Keys**” – съдържа проект, съхраняващ логиката за генериране на административен ключ и лицензи:
 - 3.1. “DigAccess.Keys” – съдържа два статични класа – за генериране на административен ключ (наречен в проекта “MasterKey”) и лицензионен ключ.
- 4. “**Services**” – съдържа проекти, свързани с бизнес логиката на контролния панел
 - 4.1. “DigAccess.Services” – състои се от класове, съдържащи бизнес логиката на функционалностите за потребителите;
 - 4.2. “DigAccess.Services.Interfaces” – състои се от интерфейсите, имплементирани от класовете в т. 4.1.
- 5. “**ViewModels**” – съдържа проект, свързан с презентационните класове
 - 5.1. “DigAccess.Models” – състои се от класовете, които биват връщани като резултат от изпълнението на методите на класовете от бизнес логиката, както и биват подавани на последните като параметри. Чрез своите свойства, те пренасят само необходимата информация между отделните компоненти на приложението.
- 6. “**DigAccess.Web**” – това е основният и стартиращ проект. Съдържа презентационната логика, контролерите и логиката за влизане/регистрация на потребители.

КОМУНИКАТОР

Комуникаторът представлява уеб-базиран интерфейс (API), чрез който се осъществява връзката между настолната програма за незрящи и базата данни. Основните задачи на комуникатора са:

1. Проверка за валидност на лиценз за „Дигитална достъпност“ и активация на такъв;
2. Извличане на потребителските идентификатори от БД;
3. Извличане на включените потребителски функционалности и техните лицензирани ключове (ако се изискват такива).

АРХИТЕКТУРА НА ПРОЕКТ

За реализирането на контролния панел, са направени 6 проекта, 5 от които са от тип “ClassLibrary” и един главен от тип “Asp.Net API”. Разделени са по предназначение в папки по следния начин:

1. **“Data”** – съдържа проекти, свързани с базата данни:
 - 1.1. “DigAccess.Data” – състои се от класовете, които се използват като модели (Entities) за базата данни;
 - 1.2. “DigAccess.DbContext” – състои се от основния за Entity Framework клас “DbContext”, който съдържа информация за таблиците, както и техните свойства (външни ключове, ограничения, стойности по подразбиране). Съхранява се и папката с миграции;
2. **“Services”** – съдържа проекти, свързани с бизнес логиката на контролния панел
 - 2.1. “DigAccess.Services” – състои се от класове, съдържащи бизнес логиката на функционалностите за потребителите;
 - 2.2. “DigAccess.Services.Interfaces” – състои се от интерфейсите, имплементирани от класовете в т.2.1.
3. **“ViewModels”** – съдържа проект, свързан с презентационните класове
 - 3.1. “DigAccess.ViewModels” – състои се от класовете, които биват връщани като резултат от изпълнението на методите на класовете от бизнес логиката, както и биват подавани на последните като параметри. Чрез своите свойства, те пренасят само необходимата информация между отделните компоненти на приложението.
4. **“DigAccess.Application”** – това е основният и стартиращ проект. Съдържа контролерите, както и конфигурационните настройки.

„ДИГИТАЛНА ДОСТЪПНОСТ“

„Дигитална достъпност“ представлява настолното приложение, което се използва от незрящите лица. За да осъществява HTTP заявки към комуникатора, е създаден клас “WebApiCommunicator”, който чрез стандартният за C# клас HttpClient, се обръща към комуникатора и взима необходимата информация. При активирането на лиценза, се взимат идентификаторите на незрящия потребител и неговия администратор и се записват в Properties.Settings.

Забележка:

1. С цел запазване на авторските права, част от настолното приложение не е публикувано в GitHub.
2. За да се изпълни приложението правилно, е необходимо то да бъде стартирано едновременно с комуникатора.
3. Демонстрация на настолното приложение:
https://www.youtube.com/watch?v=WQZwGC_HMFo&list=LL&index=18&t=226s

За повече информация относно работата на настолното приложение, моля вижте документацията на приложението, наименована „документация – Диг. Достъпност“, намираща се на директорията на проекта.