murfffi

March 2016

Анализ на общинското присъствие във Facebook

**Contents**

[Глава 1. Увод 3](#_Toc462564491)

[1. Актуалност на проблема и мотивация 3](#_Toc462564492)

[2. Цел и задачи на дипломната работа 4](#_Toc462564493)

[3. Очаквани ползи от реализацията 6](#_Toc462564494)

[4. Структура на дипломната работа 7](#_Toc462564495)

[Глава 2. Преглед на съществуващите разработки в анализа на социални мрежи 8](#_Toc462564496)

[1. Основни дефиниции 8](#_Toc462564497)

[2. Дефиниции на присъствието на общини във Facebook 9](#_Toc462564498)

[3. Съществуващи решения и подходите и методите използвани в тях за разрешаване на проблемите 10](#_Toc462564499)

[Глава 3. Анализ на изискванията към решението 13](#_Toc462564500)

[1. Концептуален модел 13](#_Toc462564501)

[2. Основни потребителски (функционални) изисквания 14](#_Toc462564502)

[3. Не-функционални изисквания 15](#_Toc462564503)

[4. Процеси в развитието на системата 16](#_Toc462564504)

[5. Работни бизнес процеси 17](#_Toc462564505)

[Глава 4. Проектиране на системата 18](#_Toc462564506)

[1. Използвани технологии, платформи и методологии 18](#_Toc462564507)

[2. Обща архитектура на системата 18](#_Toc462564508)

[1. Слой за съхранение на данни - Data Layer 19](#_Toc462564509)

[Сървър – Microsoft SQL Server и Azure DB 19](#_Toc462564510)

[Проектиране на системните аспекти на схемата на базата данни 21](#_Toc462564511)

[2. Слой за концептуалния модел на обекти и логика (Logic Layer) 22](#_Toc462564512)

[3. Автентикация на потребители 24](#_Toc462564513)

[4. Проектиране на уеб услуги (Web Services) в системата 28](#_Toc462564514)

[5. Компоненти на потребителски уеб интерфейс 30](#_Toc462564515)

[ASP.NET 30](#_Toc462564516)

[ASP.NET Dynamic Data 31](#_Toc462564517)

[Глава 5. Реализация, тестване и внедряване на системата 33](#_Toc462564518)

[6. Реализация на модулите 33](#_Toc462564519)

[1. Диаграми 35](#_Toc462564520)

[Използвана литература 37](#_Toc462564521)

[Приложения 40](#_Toc462564522)

[2. Индекс на фигурите 40](#_Toc462564523)

# Глава 1. Увод

## Актуалност на проблема и мотивация

Фейсбук (на английски Facebook) е най-голямата онлайн социална мрежа в света, която поддържа повече от 70 езика. Достъпен от 2004 г. насам, днес той има над 223 милиона потребители в Европа и постави рекорд в края на месец август тази година с над 1 милиард активни потребители за ден (Matney). Повече от 70% от потребителите на тази социална мрежа констатират, че я посещават поне веднъж дневно като над 45% от тях влизат във Фейсбук по няколко пъти на ден. Това е една от социалните мрежи с най-лесен и достъпен начин за представяне на информация и комуникация (Sarah Morse).

„Мощната глобална комуникация е започнала. Чрез интернет, хора откриват и изобретяването на нови начини за споделяне на съответните знания с ослепителна скорост. Като пряк резултат, пазарите стават все по-умни, ... и по-умни по-бързо от повечето компании.” Така започва Манифестът Клутрейн (The Cluetrain Manifesto: The End of Business as Usual) създаден през далечната 1999г. Когато четем твърденията представени под формата на тезиси, повечето от тях вече не са революционни, а част от реалността. Компаниите вече са „по-умни“ и вземат присърце първият и може би един от най-важните тезиси в труда на Rick Levine, Christopher Locke, Doc Searls и David Weinberger, а именно: „Пазарите са разговори“ (Петков). Те използват Фейсбук по възможно най-добрия начин, за да комуникират с настоящите и бъдещите си клиенти и да рекламират себе си в най-добра светлина. Освен комуникация, бизнес страниците създадени в тази социална мрежа предлагат на компаниите 1.71 милиарда активни потребители (Number of monthly active Facebook users worldwide as of 2nd quarter 2016 (in millions)), които могат да се превърнат в потенциални клиенти, намалява разходите за маркетинг на компаниите, не много скъпи целеви реклами, статистика относно харесванията, въздействието на постовете и ангажираността на потребителите към конкретна бизнес страница. Фейсбук дава възможност резултатите в търсенето на отрасъл или компания, представени със страница в тази социална мрежа да излизат на по-предни позиции, като дори корпоративните сайтове да нямат конкретно мобилно приложение, то Фейсбук страницата им ще изглежда винаги добре за всички интернет потребители, независимо какво мобилно устройство използват. (Scott Ayres)

Тези функционалности нареждат Фейсбук на едно от първите места сред социалните мрежи и е съвсем логично желанието за присъствие не само на частния бизнес, но и на държавните институции във Фейсбук. Социалната мрежа предоставя всички необходими средства за представянето, информирането и обратната връзка на тези институции със заинтересованите Фейсбук потребители.

## Цел и задачи на дипломната работа

В тази работа ще се разгледа в частност общината като държавна институция, използваща Фейсбук и нейните методи за представяне в социалната мрежа.

Въпреки, че това не е държавна политика, някои от българските общини имат присъствие във Фейсбук под една или друга форма. Не винаги обаче то е на нивото, което позволява лесно намиране на страницата на общината, информацията за общината и проектите или инициативите, които тя организира. Обратната връзка със заинтересуваните потребители и диалогът, който социалната мрежа позволява да бъде проведен с тях, са средства, които биват подценявани и не достатъчно развити в присъстващите в Фейсбук общински страници.

Като източник на горните твърдения и основа за написването на този труд е взета статията на доц. Камен Спасов и Магдалена Младенова – публикувана в информационния бюлетин на националното сдружение на общините в Република България. (Спасов)

В нея са изнесени данни, че към 2014г. от 264 общини едва 96 (36%) от тях имат някакво присъствие във Facebook. Това присъствие обаче не винаги се осъществява под правилната форма: оказва **се**, че 23 от общините (почти една четвърт от общия брой, проявили инициатива) нямат централна официална страница, а присъствието им се осъществява чрез личен профил (на самата община или нейния кмет), страница на подзвено/дирекция на общината, група или независима страница.

В някои от случаите това представяне вероятно се дължи на неразбиране на начина на функциониране на Facebook – когато общината има създаден „личен профил“ вместо страница. Служител регистрира профила, но започват да възникват въпроси – какви са рождената дата и пола на общината? Неспособността да се разграничи общинска от лична информация води до комичност в някои профили: според Facebook община Струмяни „се интересува от мъже и жени,“ а община Хаджи Димово е „безразлична“ към политиката.

В други случаи има налична страница на дирекция или подзвено на общината, но не и официална централна страница на общината.

Когато кметът на дадена община има профил във Facebook, сайтът препраща към него – факт, показващ потенциална неустойчивост, тъй като вместо да гради нещо, което ще остане и при следващите управляващи, кметът избира да поддържа единствено своята собствена страница.

Този анализ разглежда само общините със страници, освен когато изрично е уточнено друго. Въпреки че създаването на личен профил също е проява на инициатива, личният профил не отговаря на критериите ни за уместно общинско присъствие във Facebook, а и редица показатели могат да бъдат измерени само за страниците (например броят харесвания).

Целта на дипломната работа е анализ на присъствието на общините във Фейсбук и на това как те могат да станат по-популярни в тази социална мрежа, как да извлекат максималната полза от представянето си, къде се намират в сравнение с други общини по света и какво още могат да направят за подобряване на връзката община – население.

Анализът е базиран на важни характеристики, предоставени от Фейсбук като ще бъде подробно изследвана всяка от тях и какво е нейното значение за успех сред потребителите. На базата на тези характеристики, ще бъде разгледано присъствието на българските общините във Фейсбук към този момент, сравнено с минали периоди и съпоставено с общини на други държави по света.

Различни анализи, сравнения и заключения на базата на въведени данни, ще се реализират чрез единна система, предоставяща възможност на всеки да попълва информация за дадена община на базата на представянето й във Фейсбук.

Задачи, произтичащи от целта:

* Създаване на система, която ще предоставя възможност на всеки потребител да въвежда предварително дефиниран формат на информация за намерена община във Фейсбук, ще има възможност да прави търсения на база въведената информация за общините до момента и да сравнява показателите на всяка една от тях.
* Създаване на интерфейс за наблюдение на тенденциите в присъствието във Фейсбук на общините като цяло и за сравняване и анализ на резултатите.
* Системата ще има административна част, в която ще се даде възможност информацията попълнена за дадена община да бъде проверена преди тя да бъде използвана за целите на търсене и анализ.
* Системата ще има база от данни, в която ще се съдържа въведената и проверена за вярност информация, както и ще съдържа методи, по които да се проверява дали информацията е в правилния формат.
* Намиране и въвеждане на първоначална информация за анализа(към 2014г.)
* Обновяване на информацията(към 2016г.)
* Анализ на промените и на тенденциите в представянето на общините във Фейсбук

## Очаквани ползи от реализацията

Очакваните ползи от реализацията са в два аспекта:

1. Публичността на проекта и лесният достъп до информация да помогне за по-задълбочени анализи в тази област
2. заключенията от анализите направени чрез тази система ще дадат възможност да се подобри процесът по търсене, представяне и взаимодействие между общините и гражданите – потребители на Фейсбук. Систематизираната информация предоставена в приложението на тази дипломна работа ще даде възможност на общините да анализират сегашното положение и да проследят тенденциите в областта и следват добрите практики произлезли от анализите.

Освен ползите за гражданите, всяка отделна община ще добие възможността да подобри и получи повече полза от присъствието си в Фейсбук като:

* Увеличи популярността на своите Фейсбук страници като приложи подобрения, които системата показва като характерни за страници в тази област. Например администратора на социални мрежи на общината може да използва интерфейса за наблюдение на тенденциите за да определи, че има корелация между активността на посетителите и наличието на информация за географското положение на община, което би показвало че този детайл е важен за гражданите.
* Увеличи активността на потребителите, което подобрява колаборацията между гражданите. Активността на потребителите зависи от правилно използване на инструментите на Фейсбук за кратки новини, харесвания, снимки и други. Чрез възможността за следене на връзките между постове, харесвания и споменавания, системата позволява на служителите на общините да определят кои практики водят до най-голяма потребителска активност.

## Структура на дипломната работа

TODO:

# Глава 2. Преглед на съществуващите разработки в анализа на социални мрежи

Анализът на присъствие в социални мрежи е активна област на изследване. Публикувани са множество трудове от различен вид и са установени основни понятия с общоприети дефиниции (Wasserman and Faust).

## Основни дефиниции

* Социална мрежа

Социална структура, съставена от хора и организации, заедно с взаимодействията между тях се нарича социална мрежа. Взаимодействията могат да бъдат лични – приятелство, брак - родови или служебни отношения, включително между общинска власт и граждани на общината (Wasserman и Faust).

* Онлайн социална мрежа (Social Network Service, SNS)  
  Онлайн платформа, която се използва от хора и организации, за изграждане на социални мрежи, се нарича онлайн социална мрежа (Boyd and Ellison). Примери за такива онлайн платформи са Фейсбук (на англ. Facebook) – най-популярната онлайн мрежа с над 1 милиард потребители (Matney), Linkedin – мрежа, който се фокусира на служебни и бизнес отношения и Instagram – онлайн мрежа, която набляга на създаването на връзки чрез споделянето на социална мултимедия. Други популярни социални мрежи са Twitter, Youtube, Google+, Instagram и Pinterest.
* Присъствие в онлайн социална мрежа

Присъствието в онлайн социална мрежа се изразява на първо място чрез наличието на профил, с който човекът или организацията се представя на другите членове на мрежата. Профилът включва не само лични данни като име, националност, местообитание и данни за контакт, но и различни видове социална медия като текстово представяне на члена на социалната мрежа, снимки, видео и географски координати и карта, особено в случая на местни организации като общини. Втора неизменна част от присъствието в онлайн социална мрежа са връзки, предоставени от платформата, като:

* + харесване (на англ. „like”) на други членове и тяхното присъствие,
  + участие в групи по интерес, обикновено при еднолични членове
  + участие в категория на организация.

Всички онлайн връзки показват отношения в социалната мрежа в реалния свят и спомагат за пълното присъствие на човека или организацията в социалната мрежа.

* Инструменти за присъствие на организация в онлайн социална мрежа  
  Освен гореописаните начини за изразяване на присъствие в социална мрежа, някои платформи дават специални инструменти на организации, за да изграждат и поддържат връзки със граждани, клиенти и други заинтересовани лица. Тези онлайн мрежи са най-добрите места, където общините могат да има присъствие. Например Facebook дава възможност организациите да направят страница (page) вместо профил на човек. Така не се задават лични данни като пол и възраст, които нямат отношение към една организация, а вместо това въвеждането и откриването на информация, която е важна за гражданите или клиентите като адрес, работно време и контакти е по-лесно (Facebook Inc.). Друг пример е Twitter, който дава възможност на организациите на имат удостоверен (на англ. verified) профил и да анализират реакциите на свързаните хора, наречени последователи. Youtube дава възможност на общини и други организации лесно да публикуват видео в поредица наречена „канал“. Накрая, Pinterest и Instagram дават аналогична възможност като Youtube, но със снимки. TODO citation.

## Дефиниции на присъствието на общини във Facebook

Тъй като фокусът на изследването е в присъствието на общини в онлайн социалната мрежа Facebook, ще дефинираме понятията, които образуват това присъствие.

* Профил (profile) – личен профил на човек, свързан с община. Обикновено това са кметове и кандидат-кметове. Профилите включват лични данни на хора като пол, възраст, интереси и други. Някои общини погрешно създават лични профили на общината като държавно учреждение, вместо да използват страница на организация.
* Страница на организация (page) – Страниците са начина, по който организации се представят в Facebook. В тях няма лична информация за хора, а информация за организацията като цяло. Една страница може да се управлява от един или няколко души (Facebook Inc.) Понякога се ползва терминът фен страница (fan page), за да се наблегне, че страницата е начин ,по който хора, които са свързани с организацията – в случая граждани на община – да са запознати със събития за тази организация и да изразят реакциите си към тези събития.
* Публикация (post) – Освен гореописаната обща информация за страница на община(организация), съдържанието на страницата е серия от публикации с различно съдържание и цел. Някои са новини за общината, а други – обявления за начало на граждански инициативи. Медията на постовете може да е текст, снимки или видео.

## Съществуващи решения и подходите и методите използвани в тях за разрешаване на проблемите

Намерени са много разработки, които изследват присъствието и активността на общините в социалните мрежи, както и ангажираността на потребителите – гражданите към тях.

В книгата “Социалната медия и локалните управления – теория и практика“ (Sobaci) са публикувани няколко статии разглеждащи проблема.

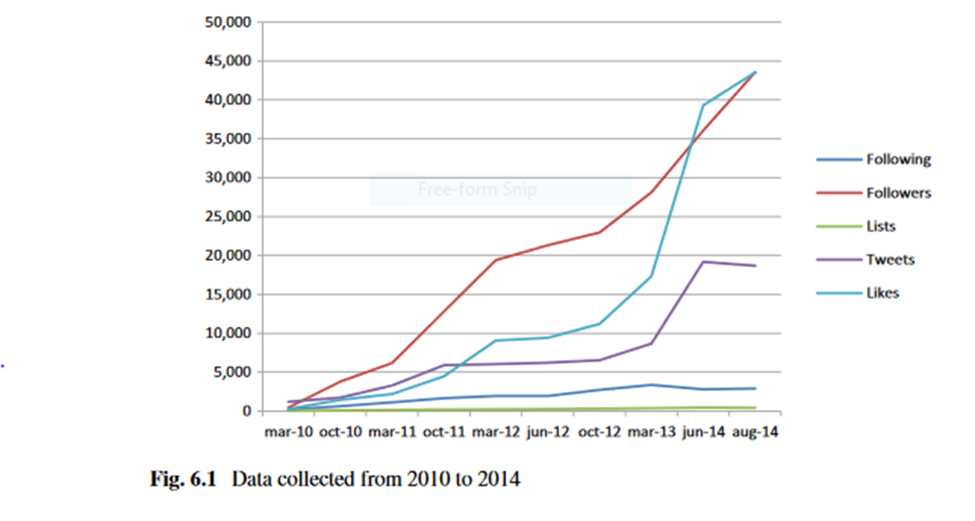
Първата разглежда използването на Facebook в западно европейските местни управления. За целта се разглежда информация събрана за 75 общини от 15 от страните най-рано присъединили се към Европейския съюз. Избрани са точно определи страни като са направени проучвания, че 10 от тях представляват 17,8% от Facebook потребителите на Европейския съюз. Резултатите от направеното проучване показват, че 73% от общините участващи в проучването имат официална Facebook страница през октомври 2012 като страните в южната част на разглеждания регион водят с 80% за разлика от тези в северната, където представянето им във Facebook e 71%.

По отношение на активността, общините в страните разположени по на юг в региона са два пъти по-активни от тези на север като създават постове средно по 2.5 пъти на ден(в работно време). За сметка на това тенденцията, която се забелязва от това проучване е, че в северния регион Facebook страниците на общините имат седем пъти повече фенове(98 523), за разлика от тези в южния(14 248). (Sobaci 59-79).

Други статии разглеждат тенденцията за нарастване на използването на Facebook от страна на общините през годините. В статия на Карън Мосбъргър от 2013г. (Mossberger) са разгледани 75 от най-големите градове в САЩ в периода 2009-2011г.. Резултатите показват, че от 13% през 2009г.. близо 87% през 2011г. използват и имат създадена страница във Facebook.

Една от публикациите, които също разглеждат тенденциите на използване на Facebook от общините през годините, но този път в Мексико.

Информацията събирана в периода от 2010 до 2014 за Facebook „харесванията“ и следенето на общинските страници от потребителите на социалната мрежа показват, че за 32-те мексиканските общини предмет на това изследване, интересът към техните страници единствено расте през годините.



British Columbia

<http://www.lgma.ca/assets/Misc/Social-Media-Primer-Research-Paper.pdf>

стр. 10 и изводите за поддържането на акаунтите в зависимост от големината на общината

Италия:

<http://cdn.intechopen.com/pdfs/40816/InTech%20The_effectiveness_of_social_software_for_public_engagement.pdf>

Ontario: <http://www.redbrick.ca/assets/file/resource/Redbrick-2014-Ontario-Municipal-Social-Media-Survey-Final.pdf>

# Глава 3. Анализ на изискванията към решението

В тази глава анализираме изискванията към система за следене на присъствието на общини в онлайн социална мрежа Facebook на базата съществуващите разработки в областта. Към всяка софтуерна система има два вида изисквания функционални и не-функционални. Първите дефинират какви възможности дава системата на потребителите, а вторите определят параметрите на качество на услугата като скорост и достъпност.

## Концептуален модел

Предвид гореописаните основни дефиниции в областта на присъствието на общини в онлайн социални мрежи, концептуалният модел от типове обекти и връзки между тях трябва да бъде като показания на Фигура 1.

Фигура 1: Концептуален модел (UML диаграма)

В модела, класовете от обекти са на английски език, за да може концептуалният модел да се използва директно при дефиниране на схемата на базата данни и йерархията от програмни класове в приложението. Всеки от обектите и техните атрибути се дефинира по следният начин:

1. Municipality – Община. TODO

## Основни потребителски (функционални) изисквания

На базата на анализа в TODO дефинираме следните най-важни изисквания. В списъка по-долу, те са подредени по тип функционалност, а не по важност.

* Автентикация на идентичността на потребител – Всички потребители трябва да могат да бъдат идентифицирани, за да могат другите потребители да са информирани за източника на всяка част от информацията за общини. Най-лесният начин за идентификация е потвърдената принадлежност към голяма социална мрежа като Facebook.
* Добавяне и разглеждане на обща информация за общини – Преди да въведат информация за присъствието на община в социална мрежа, потребителите на системата трябва да могат да разгледат основната информация за общината, а именно: име, държава, брой население и уебсайт.
* Удобство в разглеждането на информацията – Цялата информация в системата трябва да е представена във форма на таблици. Всеки атрибут на елементи като публикация (post) трябва е в колона, по която таблицата може да се сортира в нарастващ или намаляващ ред. Текстови атрибути като имена трябва да се сортират лексикографски, а числовите, като брой население, трябва да се наредят по големина. За допълнително удобство при посочването на името на всеки атрибут трябва да се показва плаващ надпис с описание на значението на атрибута.
* Отворени данни – Потребителите на системата трябва да могат да експортират данните в стандартен и отворен формат. Форматите, при които съхранението на информация в дигитален вид е описано в публикувана спецификация, се наричат отворени (Orszag). Comma-Separated Values (CSV, в превод стойности разделени със запетая) е пример за такъв формат (Shafranovich).
* Интеграция с други системи – Потребителите на системата имат нужда да я използват не само като независимо приложение, а и да я включват като готов компонент в други уеб-базирани услуги като уеб сайтове и портали. Таблиците, които са показани в системата, трябва да могат да се опишат с кратък код в HTML / JavaScript за да могат други уеб приложения да ги покажат.

## Не-функционални изисквания

Изискванията, които не са свързани с предметната област на решението и нямат директна връзка със специфичните възможности предоставени на потребителите, се наричат не-функционални (Chen, Ali Babar and Nuseibeh). Тези изисквания обаче са нужни за общото качество на работа с потребителите на системата и се очакват от всички уеб-базирани платформи. Основни не-функционални изисквания са:

* Практически постоянна достъпност – Приложението трябва да бъде постоянно достъпно дори предвид това, че компютърният хардуер не може да гарантира такова постоянство. Приложението трябва да е базирано на платформа с взаимо-заменяеми елементи.
* Скалируемост – Системата трябва да продължи да бъде достъпна при голям брой потребители, но не с заделяне на голям фиксиран хардуерен капацитет. Вместо това хардуерните изисквания на приложението трябва да скалират плавно и право-пропорционално на броя едновременни потребители и обема на управляваните данни.
* Възможност за бързо откриване и отстраняване на проблеми – За целите на практически постоянната достъпност, приложението трябва да предоставя начин да се търсят проблеми чрез преглед на вътрешни оперативни събития (логове).
* За целите на скалируемостта, приложението трябва да бъде пакетирано така че да може да се инсталира на разпределена платформена услуга (Platform-as-a-Service, PaaS) в избраният език за програмиране (ASP.NET).
* Достъпност от различни устройства – Съвременните потребители използват разнообразие от устройства и уеб браузъри (Фигура 2). Потребителският интерфейс трябва да е съвместим със всички тях, за да бъде групата от потенциални потребители на системата максимално голяма.



Фигура 2: Разпределение на броя посещения от различни уеб браузъри и устройства е света (StatCounter, Август 2016):

## Процеси в развитието на системата

За да остане системата полезна тя трябва да може да се развива. Развитието е в две отношения – подобряване на интерфейса между потребителите и системата и разширяване на поддържания концептуалният модел. Нуждата от развитие налага специален вид не-функционални изисквания.

При разширяване на концептуалният модел се правят промени на схемата на базата от данни на системата, основно под формата на добавяне на колони. Важно изискване е това добавяне да е лесно – само с промяна в графичния административен интерфейс на базата данни и синхронизация на автоматично-генериран програмен код да може едновременно да се добави колона и тя да се покаже в уеб системата на крайния потребител.

От друга страна, разширяването на приложението в посока на по-удобно ползване се постига чрез промяна на програмния код. При тези промени има възможност от грешки – регресии в други части на системата. За да намали вероятността от такива проблеми, но като се запази възможността от подобрения, има изискване да се добавят автоматични тестове. За да се изпълняват лесно и да се разработват паралелно със системата, те трябва да са на същия програмен език и платформа. Тестовете не трябва да изискват връзка с външни системи, да завършват бързо и да дават недвусмислен резултат.

Автоматичните тестове се делят на два вида – тестове на отделни компоненти (unit tests) и тестове на интеграцията между компоненти (integration tests). Комбинация от двата вида тестове е нужна, за да може негативен резултат от тестовете да показва точното място, където е възникнала регресия.

## Работни бизнес процеси

# Глава 4. Проектиране на системата

## Използвани технологии, платформи и методологии

## Обща архитектура на системата

Архитектурата на системата за записване и разглеждане на присъствие в онлайн социална мрежа Facebook e от тип многослойно приложение (Ще разгледаме по-подробно отделните слоеве и следващите точки.

За лесно инсталиране слоеве 2 и 3 са пакетирани заедно в ASP.NET уеб приложение. Схема на внедряване на приложението е показана на Фигура 4 и разгледана подробно в Глава 5. Реализация, тестване и внедряване на системата.). Множеството слоеве придават изолация между презентация и модел на данни, която е нужна заради изискванията за лесно разширяване на концептуалния модел и скалируемост (Microsoft Corporation). Слоевете на системата са:

1. Слой за съхранение на данни – Data Layer
2. Слой за концептуалния модел на обекти и логика – Logic Layer
3. Слой на интерфейса с потребители и програми – Interface layer



Фигура 3: Общ дизайн на системата като многослойно приложение

Ще разгледаме по-подробно отделните слоеве и следващите точки.

За лесно инсталиране слоеве 2 и 3 са пакетирани заедно в ASP.NET уеб приложение. Схема на внедряване на приложението е показана на Фигура 4 и разгледана подробно в Глава 5. Реализация, тестване и внедряване на системата.



Фигура 4: Схема на инсталация на системата в платформата Ажур (Microsoft Azure)

## Слой за съхранение на данни - Data Layer

### Сървър – Microsoft SQL Server и Azure DB

Сървърът, който най-добре пасва на изискванията на приложението и избраната платформа Azure e Microsoft SQL Server. Този сървър най-добре скалира между безплатен локален сървър – Microsoft SQL Server 2016 Express и скалируемa database-as-a-service (DBaaS) платформа Azure DB, която е напълно съвместима с локалния вариант с много малки изключения (Microsoft Azure).

SQL Server 2016 е стандартен сървър на база данни, който напълно изпълнява ACID изискванията, които са атомарност на операциите (Atomicity), консистентост на информацията относно релации между обекти (Consistency), изолация между паралелни операции (Isolation) и устойчивост на хардуерни проблеми (Durability). SQL Server реализира стандарта ANSI SQL 99, което прави приложението сравнително съвместимо и с други реализации на системи за управление на бази данни.

Фигура 5: Архитектура на Microsoft SQL Server (Pinal)

Архитектурата на SQL Server и интерфейсите интеграцията на стандартно приложение с него е показана на Фигура 5. На най-ниско ниво, системата за следене на онлайн присъствие използва TCP/IP протокола за връзка с базата данни и ADO.NET драйверът за платформата .NET.

### Проектиране на системните аспекти на схемата на базата данни

Схемата от таблици и колони на една база от данни обикновено следва концептуалния модел на системата. В този случай няма причина това да се нарушава. Въпреки това, реализацията на базата данни има и други аспекти, които трябва да се проектират.

Основен въпрос в дизайна е дали ще се използват естествени или синтетични основни ключове (natural vs. surrogate/synthetic primary keys) за записите в базата (Blaha). В случая, избираме да ползваме синтентични ключове заради:

* по-доброто бързодействие при обединяване на записи в две таблици по ключ. Например, обединение на facebook\_post с municipality по ключ facebook\_post.municipality\_id за да се сравни броя постове за община с нейното население
* по гъвкаво управление на неточни данни. Например, грешка в името на община лесно може да се поправи без да се налага промяна в името на общината използвано като външен ключ (Blaha).

Друга задача при системното проектиране е какво да е покритието с индекси за търсене в колоните на базата данни. Трябва да се балансира нуждата от бързо търсене, което изисква повечето колони в базата да са индексирани с нуждата от бързодействие при въвеждането на данни. Ако въвеждането на един запис обновява много индекси то потребителите ще изпитат забавяне при добавяне на данни. След анализ на изискванията, беше забелязано, че почти всички търсения по стойности в колона са при обединие на таблици както в примера по-горе. По тази причина, избрахме да създадем индекси само на колоните с първични или външни ключове.

## Слой за концептуалния модел на обекти и логика (Logic Layer)

За да бъде приложението по-гъвкаво и лесно за поддръжка, Слой за концептуалния модел на обекти е базиран на обектно-релационен адаптер (на англ. Object-to-Relational Mapping, O/RM).

Oбектно-релационен адаптер е инструмент за съхранение на данни от обекти от концептуалния модел в релационна база данни като Microsoft SQL Server. Адаптерът прави това по автоматичен начин без да има нужда от програмиране освен на някои настройки. Обектният модел на такива адаптери се състои от три основни групи:

1. обектни класове от концептуалният модел,
2. елементи на релационната база данни като таблици, изгледи (views) и съхранени процедури
3. конфигурация на съотношението между обектните класове и елементите на база данни

Чрез тази информация, oбектно-релационен адаптер позволява отделяне на дизайна на базата данни от дизайна на концептуалният модел и класове на приложението. По този начин също се автоматизират така наречените CRUD операции – създаване (Create), четене (Read), променяне (Update) и изтриване на инстанции на обекти от модела на присъствие в социална мрежа.

Типичният адаптер генерира класове от предварително създадена база данни или обратното както е показано на Фигура 6.

[](http://www.entityframeworktutorial.net/Images/ORM.png)

Фигура 6: Адаптиране между класове и таблици и други елементи на релационна база дани

Има множество обектно-релационен адаптери на пазара като DataObjects.Net, NHibernate, OpenAccess, SubSonic и други. Entity Framework е продукт с отворен код от Microsoft (Entity Framework Tutorial) и според изискванията и избраната методология се използва в системата.

В документацията си Microsoft ADO.NET Entity Framework е описан по следния начин. Той е продукт за дефиниране на съответствие между обектни класове и релационни бази данни. Чрез този адаптер разработчиците на софтуер могат да използват езика LINQ (Language-INtegrated Query, на български „заявки интегрирани в езика за програмиране) за да достъпват информация от базата данни и да я манипулират под формата на силно типизирани обекти. Освен стандартната функционалност за обектно-релационен адаптер, Microsoft Entity Framework предлага услуги като проследяване на промени по модела, централизирано идентифициране на обекти, отложено зареждане на данни и транслиране на заявки между LINQ и SQL (Structured Query Language) (Adya, Blakeley and Melnik). Всичко това позволява разработката на системата за следене на присъствие в социални мрежи да се съсредоточи на специфичната бизнес логика, а не на елементарни операции с бази данни.

Entity framework може да се използва по един от три начина (Фигура 7). Първият подход се използва когато базата данни вече съществува или архитектът предпочита да използва инструменти за проектиране на бази данни за да дефинира схемата. При вторият подход, архитектът на приложението първо създава класовете, които моделират концептуалният модел на системата и на базата на него създава таблиците и колоните в тях. Накрая, третият подход представлява съчетание на първите два. При него, архитектът първо дефинира моделът на базата данни, но на концептуално ниво. От този модел след това се генерират обектните класове и таблици на сървъра на бази данни. Интерфейсът за дизайн на база данни на концептуално ниво в интегрираната среда за разработка Visual Studio е предоставен от Entity Framework. (Entity Framework Tutorial)

В тази система ние използвахме първият вариант за дизайн, за да можем да използваме Microsoft SQL Server Management Studio при създаването на база данни със схема съвместима със съществуващите данни от избраните предишни разработки в областта.

[](http://www.entityframeworktutorial.net/Images/EF-overview.png)

Фигура 7: Видове подходи за използване на Microsoft Entity Framework за Object-Relational Mapping

## Слой на интерфейса с потребители и програми – Interface layer

### Архитектура на потребителски уеб интерфейс

Потребителският уеб интерфейс е по-голямата част от слоя на интерфейса с потребители и програми. Предвид изискванията и избраната платформа, потребителският интерфейс е реализиран с ASP.NET Dynamic Data. Технологията беше избрана на базата на анализа по-долу.

ASP.NET е унифицирана система от модел и библиотеки за разработка на уеб приложения интегрирана с платформата .NET. Тя е предназначена за създаване на динамични уеб страници и услуги. Тъй като е направена на базата на Common Language Runtime (CLR) платформата .NET, ASP.NET поддържа множество езици за програмиране –C#, Visual Basic.NET, IronPython – силно типизирана компилация, автоматично управление на паметта и наследяване.

Системата се създадена оригинално през 1992 година от инженерите на Microsoft Марк Андерс и Скот Гурти. Още отначало целта е била да се опрости създаването на разпределени приложения чрез по-структуриран и обектно-ориентиран подход и разделяне на презентацията от останалата логика на приложението (). За тази цел, ASP.NET използва така нареченият “code-behind” модел, който HTML-базираният език е в отделен файл от програмния код. ASP.NET работи тясно интегрирано с уеб сървъра за Windows машини - Internet Information Server (IIS) – за да обработва заявки на потребителите.



Фигура 11: Архитектура на ASP.NET 4.5 (Petrov)

Уеб формите са основната градивна единица при създаването на приложения с ASP.NET. Те предоставят нужната гъвкавост като представят елементите на страницата, наречени контроли (control) като обекти. Тези обекти могат да обработват събития, предизвикани от действия на потребителя като зареждане на уеб страница, натискане на бутон на мишка и други.

В допълнение на тези възможности, ASP.NET предпоставя подсистема за управление на състоянието на страница като комбинира информацията за състоянието на всяка контрола в специален атрибут наречен viewstate. Този атрибут се изпраща при всяка комуникация между клиента и сървъра като скрит елемент на страницата. Накрая, за целите на сигурността, ASP.NET се интегрира с услугите за сигурност на платформата .NET и Internet Information Server (Techopedia Inc.)

### ASP.NET Dynamic Data

Компонентът ASP.NET Dynamic Data реализира изискването за лесно разширяване на концептуалният модел на приложението. Dynamic Data автоматично генерира уеб страници и контроли на базата на Entity Framework модел на базата данни. Въпреки това, библиотеката запазва възможността за адаптиране на автоматично генерирания код към специфичните изисквания към външния вид и поведение на приложението чрез система от шаблони и контроли (MacDonald and Freeman). Вътрешната архитектура на ASP.NET Dynamic Data e показана на Фигура 12.



Фигура 12: Вътрешна архитектура на ASP.NET Dynamic Data (Microsoft Corporation)

### Автентикация на потребители

Според изискванията към решението, всички потребители трябва да могат да бъдат идентифицирани, за да могат другите потребители да са информирани за източника на всяка част от информацията за общини. Най-лесният начин за идентификация е потвърдената принадлежност към голяма социална мрежа като Facebook. Много социални мрежи като Twitter и Google+ позволяват да се използват техни уеб услуги и интерфейс за централизирано установяване на самоличността на потребител. Въпреки конкуренцията, съответната услуга на Facebook - Login for Apps – остава най-сигурният бърз и удобен начин за потребителите да се идентифицират пред произволно приложение или сайт, включително и тук описаната система (Facebook). Основна част от удобството е поради голямата възможност потребителят вече да има акаунт във Facebook.

В повече подробности, Facebook Login for Apps е бърз и лесен начин да потвърдят самоличността си пред произволен интернет сайт при използването на какъвто и да е интернет браузър и устройство. Фигура 8 показва пример за това какво потребителят ще види когато интернет сайт поиска проверка на самоличността за първи път. Услугата на Facebook е достъпна на [iOS](https://developers.facebook.com/docs/ios/login/), [Android](https://developers.facebook.com/docs/android/login-with-facebook), [уеб](https://developers.facebook.com/docs/facebook-login/web) браузъри в традиционни настолни компютри, платформата [Windows Phone](https://developers.facebook.com/docs/facebook-login/windows-phone), [традиционни](https://developers.facebook.com/docs/facebook-login/manually-build-a-login-flow) приложения и устройства като смарт телевизори.

Фигура 8: Facebook Login автентикация на смарт телефон

Интеграцията на системата за установяване на потребителска самоличност на Facebook реализира следните изисквания:

1. **Създаване на потребителски профил -** Facebook Login позволява на посетителите бързо и лесно да се запишат в системата без да се налага да конфигурират парола, която има голяма вероятност по-късно да забравят. Този прост процес увеличава възможността обикновени, вероятно случайни, посетители на уеб сайта да станат автори на нова информация за социално присъствие на общини. В бъдещето развитие на системата, това ще позволи да се използва един и същ механизъм за автентикация на множество клиентски приложения с достъп до информацията за социално присъствие.
2. **Проверена идентичност с реално име на потребителя** – Когато посетителите изберат автентикация с Facebook Login, те могат да споделят своята реална самоличност от техният профил. Публичният има профил включва тяхното истинско име, снимка и местоживеене. Тази информация, в бъдеще, ще позволи да има по-малко грешна или “spam” информацията в системата и ще повиши доверието на другите потребителите в данните. Накрая, валидираният адрес на електронна поща на потребител, който става достъпен за системата, след автентикация дава възможност за обратна връзка с потребителя и по-близка интеракция с общността от ползватели на системата.
3. **Съвместна работа с други начини на автентикация -** Facebook Login може да се използва заедно с други подходи като автентикация от други социални мрежи или обикновени акаунти с потвърждение по електронна поща или SMS. В бъдеще, това ще позволи онлайн присъствието на общините в различни социални мрежи да бъде анализирано. Електронната поща на потребителя ще се използва за да определи кой метод на автентикация е бил предпочетен от него.
4. **Потребителите контролират каква информация споделят с приложението и кога да спрат –** Доверието между всяко приложение и неговите потребители се основава на запазването на техния контрола върху лични им данни (Facebook). Facebook Login има гъвкав модел на разрешаване на достъпа до информация с над 30 параметъра. Всеки потребител вижда ясно каква информация е поискало приложението за него (Фигура 9). Освен това всеки потребител може по всяко време да прекрати достъпа към неговият профил (Фигура 10).
5. **Постепенна авторизация** - Facebook Login позволява бъдещо разширение на приложението, което би изисквало допълнителен достъп до профила на потребителите. При промяна, напълно автоматично всеки потребител ще бъде помолен еднократно да даде допълнителният достъп.



Фигура 9: Списък с атрибутите на профила, които се изискват от тестов потребител на системата Social Presence



Фигура 10: Интерфейс на Facebook за премахване на достъп на приложение (в случая Eventful) до профила на потребителя

### Проектиране на уеб услуги (Web Services) в системата

Definition - What does Web Service mean?

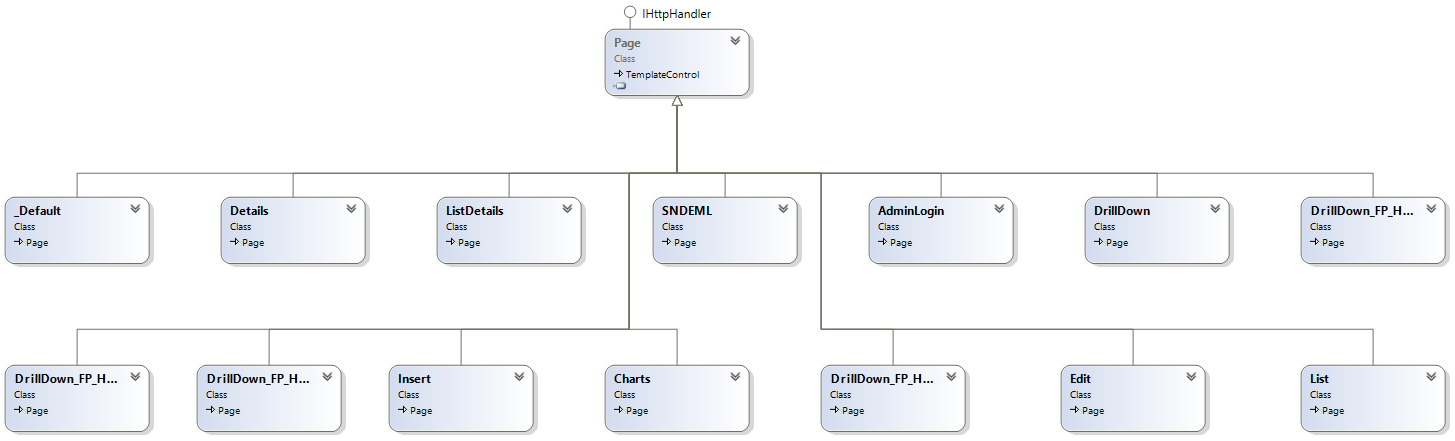
A Web service, in the context of .NET, is a component that resides on a Web server and provides information and services to other network applications using standard Web protocols such as HTTP and Simple Object Access Protocol (SOAP).  
  
.NET Web services provide asynchronous communications for XML applications that operate over a .NET communications framework. They exist so that users on the Internet can use applications that are not dependent on their local operating system or hardware and are generally browser-based.

Techopedia explains Web Service

The main advantage of a Web service is that its consumers can use the service without knowing about the details of its implementation, such as the hardware platform, programming language, object model, etc. Web service provides a loose coupling between heterogeneous systems with the help of XML messages, provide interoperability.  
  
Web services are designed to provide the messaging infrastructure necessary for communication across platforms using industry standards. Web services also use asynchronous communication to address the latency issue that arises due to requests from remote locations across the Internet. This allows the execution of background tasks for the client (such as responding to user interactions) until the actual completion of the Web service request.  
  
ASP.NET provides a framework that can be used to build Web services easily by focusing on the application logic rather than on the hardware necessary for writing infrastructure code for communication protocol or message transport. Web services created in ASP.NET can use features of the .NET framework such as caching, authentication and state management.  
  
Web service uses the ".asmx" extension with the @Web service directive (at the top of file) as per the ASP.NET application model. It can be a stand-alone application or a subcomponent of a larger Web application. (Techopedia Inc.)

 ( Stefan Bergstein, HP Software)

## Диаграми на структурата на компонентите на системата





## Графичен дизайн на потребителски интерфейс

На Фигура 13 се вижда процесът на автентикация на тестов потребител на системата – Марин, който не е администратор.

Фигура 13: Автентикация с Facebook Login в системата Social Presence.

Потребителят е натиснал бутонът Log In в горният десен ъгъл. Тъй като вече е бил използвал Facebook, той трябва само да разреши на приложението да достъпи основните му данни като име и електрона поща. Точка Автентикация на потребители в Глава 4. Проектиране на системата съдържа повече информация за всички реализирани аспекти на интеграцията.

# Глава 5. Реализация, тестване и внедряване на системата

## Реализация на модулите

## Модулно и системно тестване

## Анализ на резултатите от тестването

## Експериментално внедряване

# Глава 6. Заключение

## 

# Използвана литература

Stefan Bergstein, HP Software. *Service Oriented Architectures (SOA) Monitoring and Management with HP OpenView Operations* . 22 March 2016 r. <http://image.slidesharecdn.com/univ05ovosoaext-150828145753-lva1-app6891/95/service-oriented-architectures-soa-monitoring-and-management-with-hp-openview-operations-14-638.jpg?cb=1440773962>.

Adya, Atul, и др. „Anatomy of the ado. net entity framework.“ *Proceedings of the 2007 ACM SIGMOD international conference on Management of data*. 2007. 877-888.

Blaha, Michael. „Advanced Class Concepts.“ Blaha, Michael. *UML Database Modeling Workbook*. Technical Publications, 2013. 54-59. <https://books.google.bg/books?id=04\_XBgAAQBAJ&pg=PA59&dq=natural+vs+synthetic+keys+database&hl=en&sa=X&ved=0ahUKEwjKoKSAgarPAhWCsxQKHcTSDQgQ6AEIIDAB#v=onepage&q=natural%20vs%20synthetic%20keys%20database&f=false>.

Boyd, Danah M. и Nicole B. Ellison. „Social Network Sites: Definition, History, and Scholarship.“ *Journal of Computer-Mediated Communication* 13 (2007): 210-230. <http://dx.doi.org/10.1111/j.1083-6101.2007.00393.x>.

Chen, Lianping, Muhammad Ali Babar и Bashar Nuseibeh. „Characterizing Architecturally Significant Requirements.“ *IEEE Software* 30 (2013): 38-45. <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6365165>.

Entity Framework Tutorial. *What is Entity Framework?* 22 March 2016 r. <http://www.entityframeworktutorial.net/what-is-entityframework.aspx>.

Facebook. *Add Facebook Login to Your App or Website* . 22 March 2016 r. <https://developers.facebook.com/docs/facebook-login>.

—. *Facebook Login for Apps - Overview*. 22 March 2016 r. <https://developers.facebook.com/docs/facebook-login/overview>.

Facebook Inc. *Get Started - Facebook for non-profits*. н.д. 23 September 2016 r.

—. „Pages.“ н.д. *Facebook Help Center.* 23 September 2016 r.

MacDonald, Matthew и Adam Freeman. „Pro Asp. net 4 in C 2010.“ Apress, 2010. 1397-1436.

Matney, Lucas. *Facebook Hits New Peak Of 1 Billion Users On A Single Day*. November 2015 r. <http://techcrunch.com/2015/08/27/facebook-hits-1-billion-users-in-a-single-day/#.mkj389:Hrf0>.

Microsoft Azure. „Choose a cloud SQL Server option: Azure SQL (PaaS) Database or SQL Server on Azure VMs (IaaS).“ 2015. *Microsoft Azure.* 24 September 2016 r.

Microsoft Corporation. „ASP.NET Dynamic Data.“ 2014. *MSDN.* 24 September 2016 r.

Microsoft Corporation. „Three-Layered Services Application.“ *Patterns & practices*. Microsoft Corporation, 2002. <https://msdn.microsoft.com/en-us/library/ff648105.aspx>.

Mossberger, K., Wu, Y., & Crawford, J. *Connecting citizens and local governments? Social media and interactivity in major U.S. cities*. 2013. <http://www.sciencedirect.com/science/article/pii/S0740624X13000774>.

*Number of monthly active Facebook users worldwide as of 2nd quarter 2016 (in millions)*. 2016. <https://www.statista.com/statistics/264810/number-of-monthly-active-facebook-users-worldwide/>.

Orszag, Peter R. „Open Government Directive.“ US Government, 2009. <https://www.whitehouse.gov/open/documents/open-government-directive>.

Petrov, Ilya Ig. „Comparison of ASP.NET and Node.js for Backend Programming.“ 2016.

Pinal, Dave. „SQL SERVER – Generic Architecture.“ SQL Authority.com. 2007. Blog. <http://blog.sqlauthority.com/2007/12/08/sql-server-generic-architecture-image/>.

Sarah Morse, studioD. *How Facebook Helps Us Communicate*. 2015. <http://smallbusiness.chron.com/facebook-helps-communicate-66432.html>.

Scott Ayres. *Top 10 Benefits of a Facebook Business Page*. н.д. <https://www.postplanner.com/top-10-benefits-facebook-business-page/>.

Shafranovich, Y. „Common Format and MIME Type for Comma-Separated Values (CSV) Files.“ 2005. <https://www.ietf.org/rfc/rfc4180.txt>.

Sobaci, Mehmet Zahid, ред. *Social Media and Local Governments Theory and Practice*. Том 15. Springer International Publishing Switzerland 2016, н.д. <https://ia800208.us.archive.org/10/items/springer\_10.1007-978-3-319-17722-9/10.1007-978-3-319-17722-9.pdf>.

Techopedia Inc. *ASP.NET*. 20 March 2016 r. <https://www.techopedia.com/definition/3213/asp-net>.

—. *Web Service*. 22 March 2016 r. <https://www.techopedia.com/definition/1622/web-service-net>.

„The Cluetrain Manifesto: The End of Business as Usual.“ Rick Levine, Christopher Locke, Doc Searls and David Weinberger. *The Cluetrain Manifesto: The End of Business as Usual*. Basic Books; Reprint edition (January 2001), 1999. 190. <http://www.cluetrain.com/>.

Wasserman, Stanley и Katherine Faust. „The Social Network Perspective.“ Wasserman, Stanley и Katherine Faust. *Social Network Analysis in Social and Behaviroral Sciences*. Cambridge University Press, 1994. 1-27. <http://homepage.ntu.edu.tw/~khsu/network/reading/wasserman1.pdf>.

Zephoria Digital Marketing. *The Top 20 Valuable Facebook Statistics*. November 2015 r. <https://zephoria.com/top-15-valuable-facebook-statistics/>.

Петков, гл. ас. д-р Стойко. „Манифестът Клутрейн в епохата на социалните медии.“ 2011. <http://ebox.nbu.bg/mascom12/view\_lesson.php?id=13>.

Спасов, доц. Камен. „Анализ на общинското присъствие във Facebook.“ *Информационен бюлетин* Август 2015 r.: 20. <https://www.researchgate.net/profile/Kamen\_Spassov/publications?pubType=artifact>.

# Приложения

## Индекс на фигурите

[Фигура 1: Концептуален модел (UML диаграма) 13](file:///C:\src\social_presence\MunicipalityPresenceInFacebook.bg.docx#_Toc462564524)

[Фигура 2: Разпределение на броя посещения от различни уеб браузъри и устройства е света (StatCounter, Август 2016): 16](#_Toc462564525)

[Фигура 3: Общ дизайн на системата като многослойно приложение 19](#_Toc462564526)

[Фигура 4: Схема на инсталация на системата в платформата Ажур (Microsoft Azure) 20](#_Toc462564527)

[Фигура 5: Архитектура на Microsoft SQL Server (Pinal) 21](file:///C:\src\social_presence\MunicipalityPresenceInFacebook.bg.docx#_Toc462564528)

[Фигура 6: Адаптиране между класове и таблици и други елементи на релационна база дани 23](#_Toc462564529)

[Фигура 7: Видове подходи за използване на Microsoft Entity Framework за Object-Relational Mapping 25](#_Toc462564530)

[Фигура 8: Facebook Login автентикация на смарт телефон 26](#_Toc462564531)

[Фигура 9: Списък с атрибутите на профила, които се изискват от тестов потребител на системата Social Presence 28](file:///C:\src\social_presence\MunicipalityPresenceInFacebook.bg.docx#_Toc462564532)

[Фигура 10: Интерфейс на Facebook за премахване на достъп на приложение (в случая Eventful) до профила на потребителя 29](#_Toc462564533)

[Фигура 11: Архитектура на ASP.NET 4.5 (Petrov) 32](#_Toc462564534)

[Фигура 12: Вътрешна архитектура на ASP.NET Dynamic Data (Microsoft Corporation) 33](#_Toc462564535)

[Фигура 13: Автентикация с Facebook Login в системата Social Presence. 34](#_Toc462564536)