

УНИВЕРЗИТЕТ У БЕОГРАДУ
МАТЕМАТИЧКИ ФАКУЛТЕТ

Мастер рад

Апликација за тестирање
математичког знања у
домену друштвених мрежа

Ментор:
Проф. др Филип Марић

Студент:
Иван Милосављевић
Број индекса: 1056/2009

Београд,
септембар 2013

УНИВЕРЗИТЕТ У БЕОГРАДУ
МАТЕМАТИЧКИ ФАКУЛТЕТ
МБ. Бр. 289
ОКРАЈНОСТ

Садржај

Резиме	4
<i>Abstract</i>	5
Предговор.....	6
I. Друштвене мреже и методологије развоја софтвера	7
1. Друштвене мреже на интернету.....	7
1.1. Дефиниција друштвених мрежа	7
2. Историјат друштвених мрежа.....	9
2.1. Друштвена мрежа <i>Friendster</i>	10
2.2. Друштвена мрежа <i>MySpace</i>	10
2.3. Глобални феномен друштвених мрежа.....	12
3. Актуелне друштвене мреже.....	12
3.1. Друштвена мрежа <i>Twitter</i>	12
3.2. Друштвена мрежа <i>Google+</i>	13
3.3. Друштвена мрежа <i>LinkedIn</i>	14
3.4. Друштвена мрежа <i>Facebook</i>	14
3.5. Релације и понашање професора и ученика на друштвеним мрежама	18
3.6. Апликациони програмски интерфејси друштвених мрежа.....	20
3.6.1. Дозволе и типови АПИ-а на друштвеним мрежама	20
3.6.2. Поређење апликационих програмских интерфејса друштвених мрежа <i>Facebook, Twitter, Google+, LinkedIn</i>	22
3.6.3. Апликациони програмски интерфејси друштвене мреже <i>Facebook</i>	24
3.7. Дефинисање друштвене мреже за развој апликације	26
4. Методологије за развој софтвера	27
4.1. Агилна методологија развоја софтвера.....	27
4.2. <i>Scrum</i> методологија развоја софтвера.....	28
4.3. Обједињени развојни процес софтвера	30
4.4. Поређење методологија за развој софтвера	32
II. Развој апликације за тестирање математичког знања у домену друштвене мреже <i>Facebook</i>	36
5. Прикупљање и дефинисање захтева.....	37
5.1. Текстуална анализа захтева	37
5.2. Дијаграм захтева.....	38
5.3. Дијаграми случајева употребе	39
5.3.1. Приказ почетне стране апликације	39
5.3.2. Приказ генерисаних квизова	40

5.3.3. Решавање квиза	41
5.3.4. Предности друштвене мреже <i>Facebook</i>	42
5.3.5. Попуњавање форме за регистрацију професора	43
5.3.6. Приступ корисничком окружењу.....	44
5.3.7. Препорука пријатељу (професору)	45
5.3.8. Одабир квизова на основу критеријума.....	46
5.4. Моделовање базе података	47
5.4.1 Дијаграм базе података	47
5.5. Имплементација апликације за тестирање математичког знања у домену друштвене мреже <i>Facebook</i>	48
5.5.1. Технологије примењене у развоју апликације и развојно окружење	48
5.5.2 Опис имплементације функционалних захтева	50
5.5.3 Опис имплементације решавања изабраног квиза.....	55
5.5.4 Опис имплементације коришћења АПИ библиотека <i>Facebook</i>	59
5.6. Тестирање апликације у домену друштвене мреже <i>Facebook</i>	62
6. Дискусија	63
7. Закључак	64
Литература	65
Додатак 1.....	68
Текстуални опис дијаграма базе података	68
Текстуални опис табела базе података.....	69
Додатак 2.....	85
Регистровање <i>Facebook developer</i>	85
Креирање <i>Facebook</i> апликације.....	85

Резиме

У раду су описане друштвене мреже на интернету од њиховог настанка. Анализиране су актуелне друштвене мреже на интернету данас. Анализа актуелних друштвених мрежа имала је за циљ добијање најпогодније друштвене мреже на интернету која би била платформа, односно домен за развој апликације за тестирање математичког знања. Изабрана платформа за развој апликације за тестирање математичког знања јесте платформа друштвене мреже *Facebook*, као најактуелније друштвене мреже у Србији.

Анализиране су методологије за развој софтвера. Циљ анализе је био проналазак најпогодније методологије за развој софтвера на основу које би се радио развој апликације за тестирање математичког знања, у домену друштвене мреже *Facebook*. У раду је приказан развој апликације за тестирање математичког знања на друштвеној мрежи *Facebook* који садржи пројектну документацију и делове имплементације саме апликације. Коришћени су апликациони програмски интерфејси друштвене мреже *Facebook* при чему је апликација обogaћена друштвеним компонентама.

Кључне речи: друштвене мреже на интернету, методологије развоја софтвера, апликациони програмски интерфејс, *Facebook*

Abstract

This paper describes the social networks on the internet since the day of its inception. The paper analyzes the current social networks on the Internet today. The analysis of current social networks had a goal to get the most appropriate social network on the internet which would be the platform, or a domain for the application development used for testing of the mathematical knowledge. The chosen platform for the application development testing of mathematical knowledge is a platform of the Facebook's social network, which is the most visited social network in Serbia.

The methodology of software development has also been analyzed. The aim of the analysis was to find the most appropriate methodology for developing software that would be used for the actual development of the application for testing the mathematical knowledge, in the domain of the social network Facebook. This paper presents the development of the application for testing the mathematical knowledge on the social network Facebook, which includes project documentation and parts of the implementation, of the application itself. In the process of the application development, the application programming interfaces of the Facebook social network have been used, during which the actual application has been enriched with social components.

Keywords: social network sites, software development methodologies, application programming interface, Facebook

Предговор

Основни циљ мастер рада је израда апликације за тестирање математичког знања у домену друштвених мрежа. Садржај рада представља развој оног дела пројекта за који је био задужен бранилац мастер рада. Први део рада се односи на општи појам друштвених мрежа које чине свакодневницу великог дела популације како у свету тако и у нашој земљи уз објашњења и описе друштвених мрежа и активности корисника на њима. Рад обухвата и анализу апликационих програмских интерфејса актуелних друштвених мрежа. Поред описа и анализе друштвених мрежа, у раду се анализирају и методологије развоје софтвера. Други део обухвата развој апликације за тестирање математичког знања у домену друштвене мреже *Facebook*.

Аутор жели да изрази захвалност асистенту Мр Јелени Хаџи-Пурић и доценту др Филипу Марићу. Они су били велика подршка при изради рада, а њихово свесрдно залагање је одиграло пресудну улогу у његовом коначном изгледу. Посебну захвалност аутор дугује породици и својим најближим за неизмерну подршку пружану током читавог периода студирања.

I. Друштвене мреже и методологије развоја софтвера

1. Друштвене мреже на интернету

Друштвене мреже на интернету веома су атрактивне и заокупљују пажњу академских кругова и индустријских истраживача. Многе студије које се баве различитим феноменима интернет комуникације, а посебно оне које за предмет имају категорију „нове друштвености“, указују на чињеницу да је у питању феномен који представља алтернативу културним процесима, будући да је темељ за формирање нових или алтернативних друштвених идентитета. Поставља се питање - колико дуго су присутне и какве су друштвене заједнице раније постојале? Данас постоје многе друштвене мреже на интернету. На основу њихових дефиниција могу се класификовати различити кориснички профили. Све оне омогућавају корисницима да успоставе међусобни контакт на основу заједничких интереса (политичког погледа или активности, религије или пола и других интереса). Оне садрже комуникационе алате и средства за информисање као што су мобилна повезаност са сајтом, могућност блоговања (где блог или веблог, од енг. *web log*, *blog* чини низ хронолошки организованих уноса текста, који се приказују на профилима корисника) [1], дељење фотографија и видео снимака. Научници који се баве истраживањем друштвених медија на интернету проучавају понашање корисника на друштвеним мрежама, која им служе да уз помоћ њих проучавају праксу, културу, особине, навике и ангажовања корисника на друштвеним мрежама.

1.1. Дефиниција друштвених мрежа

Друштвене мреже су интернет сервис који омогућавају појединцима да направе свој јавни или делимично јавни профил унутар друштвене мреже, и омогућавају повезивање са осталим корисницима, као и преглед и повезивање корисника из листе корисника који су са неким другим корисником повезани. Природа и опис веза међу корисницима разликују се од типа друштвене мреже на којима је појединац направио свој профил.

Друштвене мреже су јединствене на интернету не само томе што пружају својим корисницима могућност да упознају друге, њима непознате кориснике мреже, већ им пружају могућност стварања сопствених затворених мрежа у оквиру те друштвене мреже. Због тога, у многим друштвеним мрежама, корисници не морају да остварују нове контакте са другим корисницима, већ им је примарна комуникација са људима који су део друштвене мреже тог корисника (затворен круг пријатеља).

Друштвене мреже пружају корисницима разне техничке могућности. И поред постојећих функција, главни стуб њиховог постојања је могућност виђења профила и приказивања листе пријатеља [2] који су такође корисници мреже. Профили су јединствене стране корисника. Корисници могу да их уређују и на њима опишу себе од самог рођења. При регистрацији на некој од друштвених мрежа сам формулар регистрације садржи низ питања. Профил се креира на основу одговора на та питања која укључују број година, локацију, интересовања и секцију „о кориснику“. Многе друштвене мреже охрабрују своје кориснике да поставе своје фотографије након

регистрација и креирања свог профила. Неке од њих дозвољавају проширивање профила додавањем мултимедијалног садржаја. Друге, као што је *Facebook*, дозвољавају додавање модула (“апликације”).

Видљивост профила корисника варира у зависности од природе друштвене мреже и дискреције коју жели сам корисник. Неке од њих дозвољавају да се профили њихових корисника приказују у резултатима претраге на *Google* претраживачу и тако дозвољавају сваком да види тражени профил, без обзира на то да ли особа која гледа профил корисника поседује налог на тој друштвеној мрежи или не. Друштвене мреже као што је *MySpace* дозвољавају својим регистрованим корисницима да изаберу на свом профилу шта желе да им се приказује јавно и шта желе да виде њихови “пријатељи” (енг. *Friends only*). *Facebook* има другачији тзв. стандардан приступ. Уколико су корисници део одређене групе, они могу да виде сваки други профил из те групе, осим ако сам корисник није одлучио да не дозволи доступност ових информација. Постоје варијације степена видљивости и приступа профилима корисника. Управо и разлике међу друштвеним мрежама потичу од тога како су организовани подаци корисника на одређеној друштвеној мрежи.

Придруживањем некој друштвеној мрежи, кориснику се нуди идентификација других корисника у систему друштвене мреже који по неким критеријумима могу да поседују релацију са њим. Постоје различито дефинисани односи код различитих друштвених мрежа. Тако можемо да препознамо популарне термине као што су: “Пријатељи”, (енг. *Friends*), “Контакти” (енг. *Contacts*), “Обожаваоци” (енг. *Fans*) и други. Многе од њих захтевају усмерену релацију између корисника која се назива “пријатељство”. Постоји и неусмерена релација која се може категорисати под појмом “Обожаваоци” или “Следбеници” (енг. „*Followers*”). Међутим, многе друштвене мреже везе међу корисницима категорише под појмом “Пријатељи”. Сам појам “Пријатељи” може бити варљив зато што релација међу корисницима не мора нужно значити и свакодневно пријатељство међу њима већ је то појам који је постао синоним за релације између појединаца на друштвеним мрежама.

Јавна веза међу корисницима је круцијална компонента друштвених мрежа. Листа пријатеља садржи везу ка профилу сваког од њих која дозвољава кориснику да обијаси мрежни граф направљен од листе пријатеља.

Многе друштвене мреже поседују механизам којим корисник може остављати поруке на профилима пријатеља. Ова погодност је типично остављање “коментара”, али се она разликује на друштвеним мрежама. Често постоји могућност слања приватних порука које је слично електронској пошти (енг. *e-mail*). Међутим, ови механизми нису универзално решење за све друштвене мреже.

Поред појмова профила, пријатељства, коментара и приватних порука, друштвене мреже пружају и многе друге погодности. Неке поседују дељење статуса, слика или дељење видео-записа, друге имају уграђено блоговање у себи и слање инстант-порука (енг. *chat*). Постоје чак и друштвене мреже за псе (енг. *Dogster*) и мачке (енг. *Catster*), где власници љубимаца “одржавају” профиле љубимаца.

2. Историјат друштвених мрежа

Према горе наведеним дефиницијама, прва друштвена мрежа направљена је 1997. године под називом *SixDegrees.com* [3]. На самом почетку ове друштвене мреже појединци могли су да креирају профиле и да прегледају листу пријатеља, а почетком 1998. могли су да прегледају профил сваког пријатеља. Све ове могућности постојале су у разним другим сервисима и многим комуникационим сајтовима и пре постојања саме мреже *SixDegrees*. Тако су сервиси *AIM* и *ICQ* омогућавали креирања листе пријатеља али она није била доступна другим корисницима. Још једна друштвена мрежа из тог времена је *Classmates.com* која је пружала могућност повезивања са колегама из средње школе или факултета. Њима у почетку није било омогућено да прегледају листу својих пријатеља. Зато се мрежа *SixDegrees* разликовала од осталих у то време јер је имала комбинацију горе наведених функција.

SixDegrees је настала као концепт шестог степена раздвајања који гласи: свако на свету, приближно у шест корака, може да направи ланац пријатеља, полазећи од било које особе на свету [4]. *SixDegrees* је промовисана и као алат који помаже корисницима да успоставе везу и шаљу поруке једни другима (енг. *Web of Contacts*). *SixDegrees* је бројао милионе корисника, али је због самог одржавања сервиса, боље речено, неодрживог бизниса, 2001. године сервис угашен. Оснивач ове друштвене мреже навео је разлог гашења тај да је она била веома једноставна у односу на време у ком је постојала. Још један од разлога гашења ове друштвене мреже био је проблем развијања и доступности интернета због кога корисници који су користили интернет у то време нису имали потребе да развијају мреже пријатеља који су били такође на интернету. Нека истраживања из тог времена говоре о томе да људи нису били заинтересовани за упознавање других корисника који су њима били непознати.

У периоду од 1997. до 2001. године знатно је био увећан број комуникационих алата који су имали разне комбинације корисничких профила и јавно доступних "пријатеља". Неки од њих омогућавали су кориснику да креира лични профил, професионални профил или профил за забављање. Регистровани корисници су могли да идентификују профиле пријатеља и да се повежу уз дозволу својих пријатеља. По истом принцип, почетком 1999. године, *LiveJournal* [5] је направио неусмерене релације међу странама корисника. Није било потребно успостављање усмерене релације, већ су корисници могли да изаберу одређене "пријатеље" и да прате њихове текстове. У том периоду развијају се значајне друштвене мреже и то: у Кореји (*CyWorld* [6]) и Шведској (*LunarStorm* [7]) које су имале сличне особине као што је *LiveJournal*.

Друштвена мрежа *Ryze.com* [8] појавила се на интернету 2001. године. Она је била направљена са циљем да помогне људима у пословном свету и представљала је тзв. пословну друштвену мрежу. Поред ње, у друштвене мреже сличног типа спадају: *Tribe.net* [9], *LinkedIn* [10] и *Friendster* [11] где су испреплетани лични и пословни профили корисника. Идеја која је постојала код креатора ових мрежа била је повезивање личног и пословног профила корисника и пружање услуге удруживања. Међутим, друштвена мрежа *Ryze* никада није постигла велику популарност као ни *Tribe.net*. Данашња популарна друштвена мрежа на којој се пословног типа јесте друштвена мрежа *LinkedIn*.

2.1. Друштвена мрежа *Friendster*

Friendster је значајна друштвена мрежа која је обележила почетак постојања огромне заједнице људи који се на интернету. Настала је 2002. године. Концепт на ком се заснивала је био комплементаран концепту друштвене мреже *Ryze*, а дизајнирана је као конкурент друштвеној мрежи *Match.com* [12] која је служила за проналажење љубавних партнера. Била је замишљена као мрежа другачија од класичних интернет сервиса за упознавање и њен концепт се заснивао на повезивању пријатеља од пријатеља (енг. *friends-of-friends*) како би корисници могли “лакше” да пронађу партнере у односу на упознавање странаца преко других интернет сервиса. У свом раном развоју, на друштвеној мрежи прикључило се 300 000 корисника а први корисници били су блогери – посетиоци уметничког фестивала *The Burning man* и велики број геј популације. Друштвена мрежа *Friendster* имала је у једном тренутку око 10 милиона регистрованих корисника. Из тог разлога је настали су огромни проблеми техничке природе. Сервери и базе података тешко су подносили нагли раст корисника. Са друштвене стране, корисници нису могли да раздвоје листе пријатеља. Морали су у исто време да имају, у оквиру листе пријатеља, своје шефове и колеге са посла заједно са најближим пријатељима. Ово је представљало велики проблем за кориснике. Поред свих наведених потешкоћа, *Friendster* је почео да ограничава активности својих корисника. Једно од значајних ограничења на *Friendster* је било ограничење гледања профила корисника који су били удаљени више од четири нивоа пријатеља (*friends-of-friends-of-friends-of-friends*). Ово ограничење је било контрапродуктивно. Корисници су почели да додају људе које не познају. Поред тога постојала је и опција “најпопуларнији пријатељ”, због чега су корисници и тада желели да имају што више пријатеља. У томе су највише успеха имале славне личности, као и корисници који су правили лажне профиле под именима познатих или измишљених особа. *Friendster* је у једном тренутку покренуо акцију брисања свих лажних профила и тиме је настало неповерење корисника ка овој друштвеној мрежи. Сви наведени проблеми су стварали незадовољство код корисника, па су они тражили друге друштвене мреже које су биле актуелне у то време. Неки је описују као велико разочарање у историји интернета. У децембру 2009. године *MOL Global* је купио друштвену мрежу *Friendster*. Данас представља интернет сервис за играње игара.

2.2. Друштвена мрежа *MySpace*

Почетком 2003. године појављује се све више друштвених мрежа. Поред њих, повећава се и број сервиса који имају друштвени принцип у свом функционисању. Такви сервиси су познати корисницима као сервис за дељење фотографија (*Flickr* [13]), сервис за слушање музике (*Last.FM* [14]) као и сервис за дељење видео-записа (*Youtube* [15]).

Исте године се појављује и друштвена мрежа *MySpace* [16]. Она је била популарна од самог оснивања. Њен оснивач је Том Андерсон (Tom Anderson). Он је желео да искористи проблеме друштвене мреже *Friendster*. Претпоставио је да ће корисници ове мреже тражити алтернативу. Успео је у томе и привукао је незадовољне кориснике друштвене мреже *Friendster*. Тако је друштвена мрежа *MySpace* добијала прве кориснике. Захваљујући таквом приступу *MySpace* је

забележила брзи раст. Један веома занимљив део ове друштвене мреже у њеној раној фази развијања је регистравање популарних инди-рок бендова на друштвеној мрежи које је *Friendster* избацио јер се нису придржавали прописа о профилима. Они су подстицали све друге кориснике друштвене мреже *Friendster* да пређу на друштвену мрежу *MySpace*.

Обзиром да *MySpace* у свом концепту није имала повезивање бендова са обожаваоцима, инди-рок бендови су били добродошли на мрежи. То су били бендови из региона Лос Анђелеса. Они су промовисали друштвену мрежу *MySpace* локално, тако што су на самој друштвеној мрежи продавали ВИП улазница за популарне клубове у том региону. Прихватајући идеју промоције друштвене мреже њен оснивач је контактирао и остале локалне бендове који нису били на овој друштвеној мрежи. Он им је предлагао да направе профиле својих бендова на мрежи, а за узврат им је нудио помоћ у промовисању њихове музике на друштвеној мрежи. Овакав начин промовисања друштвене мреже *MySpace* није био и једини. Ова веза је омогућавала бендовима да буду у контакту са својим обожаваоцима. Са друге стране, обожаваоци су могли да прате њима занимљиве бендове и користећи везу "пријатељ" да се повежу са њима. Осам месеци након покретања, друштвена мрежа *MySpace* је доживела "ефекат мреже" (енг. *network effect*) - експоненцијални раст корисника на мрежи. (Ефекат мреже представља појам умрежавања и креирања мрежа, а који је указао Роберт Меткалфе (Robert Metkalfе), творац Етернета. Његова формулација проистиче из области телекомуникација, али се примењује и у комуникологији, науци о Вебу, социологији, названом Меткалфов закон : "Вредност мреже расте као квадрат броја (њених) корисника", а одређује се према математичком исказу n^2 , у којем n означава број корисника. [17])

Константно су додавана нова својства уз која су корисници могли да уреде своје профиле на различите начине. Назив "*MySpace*" је проистекао из идеје да корисник добије свој простор где може да ради шта жели. Тако су корисници могли да додају *HTML* (*HyperText Markup Language*) код на своје профил стране и тиме креирају јединствен *MySpace* изглед.

Почетком 2004. године тинејџери почињу масовно да се прикључују друштвеној мрежи *MySpace*. У односу на старије (по годинама) кориснике, тинејџерима није била интересантна друштвена мрежа *Friendster*. Већина њих је желела да се повеже са својим омиљеним бендовима, други су се прикључивали друштвеној мрежи на препоруку старијих чланова породице. Када се сами прикључе, тинејџери позивају своје пријатеље да се и они активирају на мрежи. У одређеном тренутку *MySpace* мења политику и дозвољава малолетницима да буду њени корисници. Како је друштвена мрежа бележила свој раст примећивале су се три различите групе корисника и то: музичари/уметници, тинејџери и корисници које по свом развојном добу можемо да назовемо корисници средњих година (односно корисници који су завршили факултет) [*post-college urban social crowd*]. Можемо приметити да чланови последње две групе корисника (тинејџери и корисници средњих година) нису имали директне контакте једна са другом, односно да им је једина веза била преко прве групе корисника (музичара/уметника).

У јулу 2005. године новинска корпорација *News Corporation* купила је *MySpace* за 580 милиона долара и тиме закупила масовну медијску пажњу. Отворило се питање безбедности на друштвеној мрежи. Бележиле су се серије интеракција сексуалне природе одраслих и малолетника што је изазвало панику и реакцију у јавности. Морална паника брзо се проширила у вези сексуалних предатора који се

налазе на друштвеној мрежи иако су истраживања показала да је забринутост преувеличана (истраживања су показивала да је 1 од 7 тинејџера примило узнемирујуће и нежељене сексуалне поруке на друштвеној мрежи, и да је 9% тих порука дошло од корисника који су имали преко 25 година).

2.3. Глобални феномен друштвених мрежа

Док је друштвена мрежа *MySpace* заокупљала главну медијску пажњу у Америци и шире, сам феномен друштвених мрежа ширио се и добијао је на значају свуда у свету. Друштвена мрежа *Friendster* била је популарна на Пацифичким острвима; *Orkut* је била примарна у Бразилу пре невероватног раста у Индији; *Mixi* је била популарна у Јапану; *LunarStorm* у Шведској; *Hvves* у Холандији; *Grono* у Пољској; *Hi5* је обухватала кориснике Јужне Америке, Северне Америке и Европе; *Bebo* је постала веома популарна у Великој Британији, Новом Зеланду и Аустралији. Комуникациони алати су заузели примарно место у друштвеним мрежама. Све више је било заступљено популарно писање порука међу корисницима, односно ћаскање (енг. *chatting*). Кинески QQ сервис за размену порука међу корисницима (енг. *instant messaging service*) прерастао је у огромну друштвену мрежу када је додао могућност уређивања профила својим корисницима. *CyWorld* је у Јужној Кореји добио на важности када је направљена трансформација форума тако што су додали могућност да корисници уређују своје профиле на форуму и додају пријатеље.

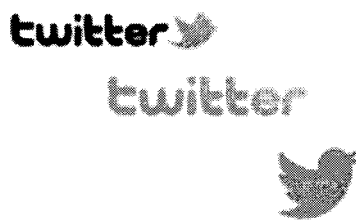
Блоговање је постало основна и веома популарна особина друштвених мрежа. Широм света разни веб сервиси су уводили алат за блоговање којим су ти сервиси попримили особине друштвених мрежа. Њиме су привлачили велики број корисника и тиме ширили свој домен пословања.

3. Актуелне друштвене мреже

У данашње време живот без друштвених мрежа готово да је незамислив. Људи свакодневно проведу више сати за рачунаром и на интернету, односно на некој друштвеној мрежи. Готово свуда у свету постоје друштвене мреже на којима се људи дописују једни са другима, деле своја интересовања, мишљења, радост и тугу. Најпознатије друштвене мреже данашњице биће описане у наставку рада.

3.1. Друштвена мрежа *Twitter*

Друштвена мрежа *Twitter* [19] је основана 2006. године, а њена основа је писање микроблогова (енг. *microblogging*). Кратке поруке (текстуалне поруке дугачке максимално 140 знакова) које корисници ове друштвене мреже могу објављивати и читати називају се *tweets*. По њима је *Twitter* и добио име. Због сличности са SMS-концептом (*Short Message Service*) у покретним телекомуникацијама *Twitter* се често назива "SMS интернета".



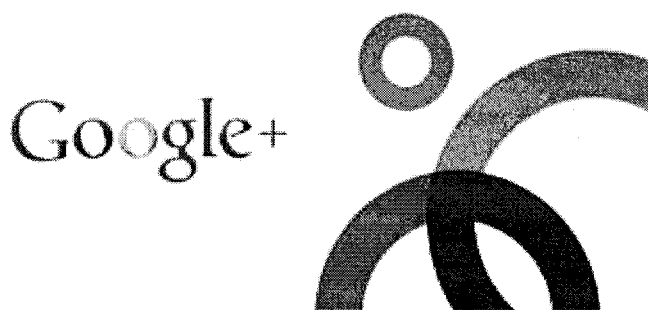
Слика 1. Логотипови *Twitter*-а од 2006 до 2012. године

У почетку није било ограничења дужине твитова. То је довело до тога да су корисници почели да пишу дугачке поруке које су оптерећивале и сервис и друге кориснике и уништавале саму идеју *Twitter*-а као сервиса за остављање кратких порука. Након неког времена постављено је ограничене дужине твита на 140 знакова јер је дужина SMS поруке била 160 знакова, па је тих 20 знакова разлике остављено за корисничко име.

Данас *Twitter* броји 500 милиона активних корисника широм света [20] који генеришу више од 340 милиона твитова дневно, односно 1,6 милијарди твитова недељно.

3.2. Друштвена мрежа *Google+*

Друштвена мрежа *Google+* [21] или *Google plus* је друштвена мрежа која је настала јуна 2011. године [22] од стране *Google Inc.* Иако је релативно млада она је забележила успех у свом домену постојања. Броји 500 милиона регистрованих корисника. На овој друштвеној мрежи, пријатељи се групишу на основу интересовања а не по томе одакле их корисник познаје. Интересантна новина која се налази на овој друштвеној мрежи јесу кругови (енг. *circles*) која омогућује груписање пријатеља. Кругови могу да се организују по целинама, односно по томе које информације корисник жели да подели са одређеном групом пријатеља. То значи да, уколико корисник жели да са одређеном групом пријатеља подели вести о спорту, онда би корисник требао да направи посебан круг на овој мрежи и да се у том кругу налазе његови пријатељи које интересује спорт. Уколико корисник жели да подели одређене информације само блиским пријатеља, на пример слике са одређене прославе, онда може да направи нови круг пријатеља у коме би требало да се налазе само блиски пријатељи.



Слика 2. Концептуални логотип *Google+*-а

Поред додавања пријатеља, на друштвеној мрежи *Google+* корисник може да дода слике, видео-исечке, линкове. Корисницима је омогућено да постављају своје статусе. У специјалном случају, подржана је забрана дељења одређеног статуса на овој друштвеној мрежи (енг. *disable reshare*), слика или других приватних детаља. Друштвена мрежа *Google+* поседује могућност размена инстант-порука (енг. *chat*) као и видео-конференција (енг. *hangout*).

Главни проблем ове друштвене мреже представља мала активност регистрованих корисника. Она је забележила нагли раст у односу на период настанка. Међутим, у односу на број регистрованих корисника веома мали проценат њих активно користи ову друштвену мрежу.

3.3. Друштвена мрежа *LinkedIn*

Друштвена мрежа *LinkedIn* је настала 2002. године. Њен креатор је Реид Хофман (Reid Hoffman) који је уједно и суоснивач ове друштвене мреже. Маја 2003. године званично је и почела са радом на интернету и у првих месец дана окупила је четири хиљаде и пет стотина чланова. Ова друштвена мрежа намењена је људима који желе да се повежу са другим појединцима и организацијама у пословном виртуелном свету. Најважнији део профила једног корисника је каријера, послови, едукација и друге особине и вештине корисника које дефинишу професионални профил. Ова друштвена мрежа је веома корисна за менаџере и директоре компанија. Многи менаџери који управљају људским ресурсима (енг. *human resource HR*) имају могућност претраживања великог броја потенцијалних кандидата за запослење. Међутим, основни профил на овој друштвеној мрежи има ограничене могућности. Уколико корисник жели додатне функције онда мора да их плати. Занимљиво је да корисници који свакодневно ажурирају свој профил на овој друштвеној мрежи, уколико траже нови посао, када конкуришу за посао, не шаљу своје биографије већ само свој профил на овој друштвеној мрежи. Због тога се профил на овој друштвеној мрежи назива виртуелна биографија (лат. *curriculum vitae CV*) која описује професионалну каријеру и особине појединца.



Слика 3. Логотип друштвене *LinkedIn*

3.4. Друштвена мрежа *Facebook*

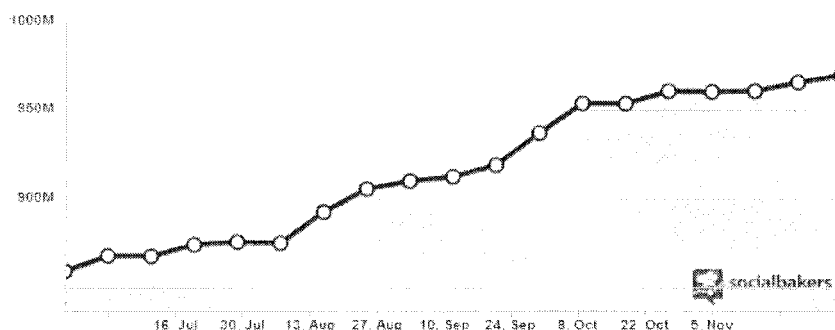
Друштвена мрежа *Facebook* [23] настала је 2004. године. Њен оснивач је Mark Zuckerberg. Он је био студент Харварда који је у почетку са још тројицом колега покренуо друштвену мрежу под називом *thefacebook.com* намењену само студентима Харварда. За регистрацију на ову друштвену мрежу било је потребно поседовање *e-mail* адресе са доменом *harvard.edu*. Временом је на друштвеној мрежи омогућена регистрација и студената са других универзитета, али су они такође морали да се

региструју са *e-mail* адресе свог универзитета. 2005. године назив ове друштвене мреже је промењен у *facebook.com*. У септембру исте године на друштвеној мрежи је омогућено регистровање ученика средњих школа, потом и пословних људи у великим компанијама, док је данас доступна свим корисницима старијим од 12 година. *Facebook* је данас најпопуларнија друштвена мрежа која броји 963 812 360 корисника (слика 5).

facebook

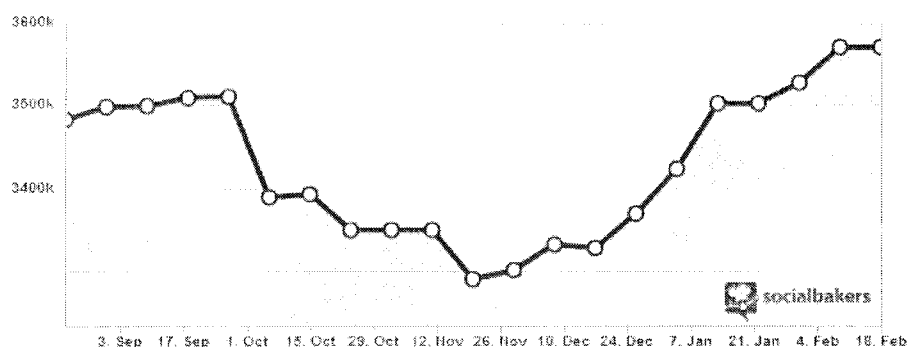
Слика 4. Логотип друштвене мреже *Facebook*

До краја 2012. године ова друштвена мрежа је забележила константан раст корисника (слика 5).

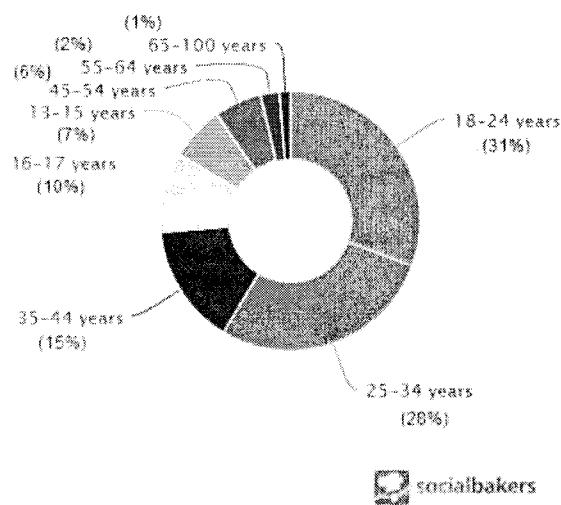


Слика 5. Развој *Facebook*-а до јануара 2013. године [24]

Веома занимљива информација је да се на овој друштвеној мрежи регистровало више од 50% становништва Србије [25] чиме се наша земља истиче по броју регистрованих корисника и ставља је на 46. место у свету по броју регистрованих корисника. По статистици из фебруара 2013. године укупан број регистрованих корисника у Србији износи 3 570 380. Од укупног броја регистрованих корисника највећа група по броју година је у распону од 18. до 24. године и износи 1 094 900 корисника, затим прати је група корисника у распону од 25. до 34. године (слика 7.). Од укупног броја регистрованих корисника у Србији 55% корисника су мушкарци док 45% корисника чине жене.



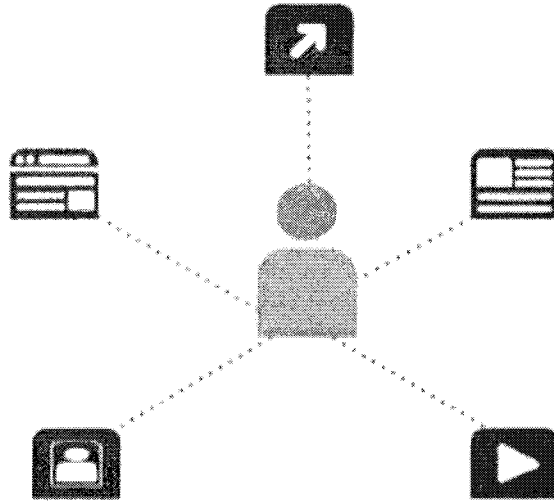
Слика 6. Регистровани корисници на *Facebook*-у у Србији у до фебруара 2013. Године



Слика 7. Распон година корисника на *Facebook* у Србији до фебруара 2013. године

Постоји огроман број питања која се могу поставити везана за број регистрованих корисника на овој друштвеној мрежи у нашој земљи и у свету. Она питања којима се бави овај рад јесу питања која се односе на кориснике који су присутни на друштвеној мрежи *Facebook* са циљем унапређења личног образовања.

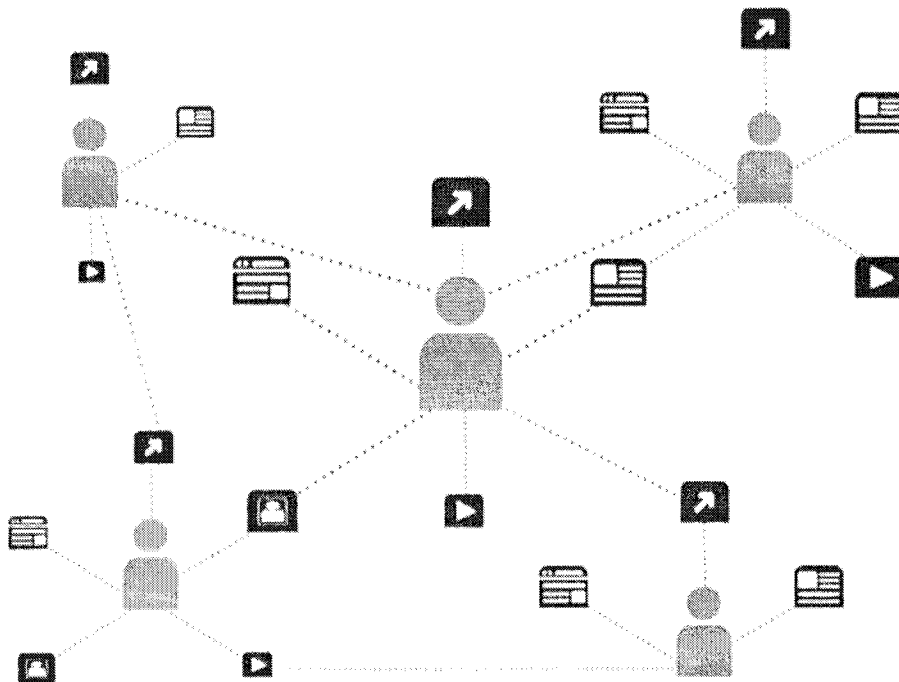
Да бисмо лакше посматрали друштвену мрежу *Facebook* и анализирали њене предности и мане дефинисаћемо је као скуп веб страна. Свака страна представља профил корисника. Оне омогућавају корисницима да међусобно комуницирају тако што могу да уређују информације о себи, пишу статусе, размењују видео-записе и слике, склапају пријатељства са другим корисницима ове друштвене мреже (слика 7).



Слика 7. Профил корисника на *Facebook*-у

Комуникација између корисника остварује се на више начина: слањем приватних порука у оквиру друштвене мреже, разменом инстант-порука (енг. *chat*), остављањем порука на самом зиду корисника (енг. *wall*), играњем истих игара у домену друштвене мреже, коментарисањем слика, порука на профилу корисника, видео-записа корисника и слично.

Интеракција и комуникација међу корисницима образује тзв. мрежни граф. Мрежни граф у домену друштвених мрежа представља релације међу корисницима и садржајима корисника (слика 8.).



Слика 8. Мрежни граф корисника *Facebook* друштвене мреже

Код друштвене мреже *Facebook* могуће је подесити разне нивое сигурности и приступа корисничком профилу, односно видљивост профила корисника, о чему је раније дискутовано. Постоје различита ограничења и корисници могу да их комбинују и одреде шта други корисници виде на њиховом профилу, да подесе одређену групу корисника са одређеним нивоом приступа података и профилу корисника.

3.5. Релације и понашање професора и ученика на друштвеним мрежама

Питање које је може да се постави у овом тренутку јесте да ли међу ученицима, односно студентима треба да постоји релација ка професорима (наставници у основној школи, професори у средњој школи и на факултету) и који ниво интеракције треба да постоји у тој релацији. Јасан одговор очигледно није могуће одмах дати.

Ако би постојао прописан одређени етички кодекс на основу чега би професори могли да одреде смернице у интеракцији међу својим ученицима и тада би постојале ситуације које нису решиве и јасно дефинисане. У једном од дефинисаних правила понашања у САД наводи се:

“Професори би требало да охрабрују ученике у слободном приступу учења и проналаска новог знања. Они би требало да поставе ученицима стандарде у науци и етичка правила сопственим понашањем. Потребно је да поштују ученике, сваког понаособ, и да буду доследни у својој улози интелектуалног водича и саветника. Требало би да чине сваки разуман напор да би подстакли искрено академско понашање и напредак ученика који се одражава као истинска заслуга сваког ученика. Потребно је да поштују природну релацију између професора и ученика. Потребно је да избегавају било какву експлоатацију, узнемиравање или дискриминацију ученика. Требало би да признају значајну академску или научну помоћ од стране ученика. Требало би да штите академску слободу.” [26]

Кодекс понашања дефинисан на овај начин не постоји у Србији, а његово присуство значајно би помогло у компаративној анализи и одређивању смерница у наставку рада.

Међутим, уколико обратимо пажњу на горе наведени део етичког понашања професора у САД (могли бисмо да претпоставимо да би требало да сличан кодекс важи и за професоре у Србији) добијамо закључак да међу професорима и ученицима, односно студентима треба да постоји одређена релација на друштвеним мрежама. Она треба да буде релација као у реалном животу. Професори би требало да се понашају “професионално” на друштвеним мрежама и у складу са својим обавезама према ученицима/студентима.

Проучавајући студију која је обухватала анализу понашања ученика/студената у односу на број година на друштвеној мрежи *Facebook* [27], добијамо следеће закључке:

- Корисници који су регистровани у периоду од своје 13. године до 17. године често друштвене мреже користе за организацију свог друштвеног живота, и друштвене мреже им служе за истраживање и развој вештина у свом друштвеном животу. Друштвене мреже овој групи корисника најчешће служе и за приказивање личних

особина и других информација које имају вишеструко значење у комуникацију и размену информација међу пријатељима, док су у основној и средњој школи.

- Корисници који су регистровани у периоду од своје 18. године до 24. године друштвене мреже користе за разне друге ствари. Они проводе време и усмеравају енергију у изградњи веза ка разним интересним групама и комуникационим заједницама. Друштвене мреже служе им као мултифункционалан комуникациони алат који користе за време студирања. На њима размењују и добијају корисне информације.

- Корисници чије је занимање предавач (учитељ или професор) могу доћи у контакт са другим корисницима различитих година тако да могу добити квалитетне повратне информације од других корисника. Таква врста интеракције може да помогне предавачу да разуме своје "ученике" и од њих сазна какве све пожељне особине треба да има предавач.

Потенцијал друштвених мрежа за повећавањем интеракције и комуникације између ученика/студената и професора очигледно постоји. *Facebook*, као највећа и најактуелнија друштвена мрежа у овом тренутку, нуди такву могућност. С обзиром на то колико је тренутно корисника ове друштвене мреже присутно у нашој земљи, у раду биће приказан начин на који је могуће креирати веб апликацију у домену ове друштвене мреже користећи све њене предности.

3.6. Апликациони програмски интерфејси друштвених мрежа

Апликациони програмски интерфејс (АПИ) (енг. *Application programming interface, API*) је интерфејс који дефинише начине на који софтверске компоненте комуницирају једне са другима. Практично, у много случајева АПИ је библиотека која најчешће садржи спецификацију за структуре података, објекте, класе и протоколе који се користе за комуникацију између апликације која тражи услугу одређеном сервису коришћењем АПИ библиотеке. [28]

Популарне платформе друштвених мрежа могу да се назову сервисима идентитета. Оне све више постају занимљивије за развијање апликација и повезивања веб сајтова са друштвеним мрежама уз коришћење њихових апликационих програмских интерфејса. Оне дозвољавају апликацијама или сервисима да користе податке регистрованог корисника, уколико корисник користи такву апликацију или сервис. Кроз апликациони програмски интерфејс, апликација или сервис неког веб сајта може да се повеже са активностима корисника, може и да одређену вест објави на корисничком профилу, уколико је корисник претходно дозволио такву врсту интеракције. Данас није уопште неуобичајено то што су велике компаније, како у свету, тако и код нас присутне на актуелним друштвеним мрежама попут *Facebook*, *Twitter*, *Google+*, *LinkedIn* и другим друштвеним мрежама које им омогућавају и повећавају ангажовање, видљивост и утицај на крајњег корисника њихових производа или услуга. [29]

Апликациони програмски интерфејси на интернету последњих година креирани су као скуп сервиса који олакшавају дизајн и развој модерних веб апликација. Популарне друштвене мреже као што су *Facebook*, *Google+*, *Twitter*, *LinkedIn* дозвољавају програмерима да користе њихове библиотеке и да повезују своје веб апликације са овим друштвеним мрежама. Ове друштвене мреже пружају могућност лаког приступа деловима корисничких података који могу да се добију повезујући веб апликацију са неком од друштвених мрежа, користећи АПИ библиотеке друштвених мрежа.

3.6.1. Дозволе и типови АПИ-а на друштвеним мрежама

Ауторизација одређује права приступа одређеним ресурсима. Друштвене мреже за одређивање права приступа апликацијама или сервисима користе отворени стандард за ауторизацију (енг. *OAuth* [30]). То је стандард који „дозвољава безбедан приступ“ неком ресурсу на интернету. Код друштвених мрежа овај стандард је доста заступљен. Постоје различите верзије овог стандарда. Најновија верзија овог стандарда је 2.0. Друштвене мреже користе различите верзије овог стандарда. Поређење овог стандарда код друштвених мрежа приказано је у табели 1.

Друштвене мреже имају развијене неколико типова АПИ-а и то најзаступљенији су:

- Клијентски или *JavaScript* [31] АПИ
- Серверски или *REST* [32] АПИ

Сваки од горе наведених типова АПИ могуће је интегрисати на некој веб презентацији или апликацији. У зависности од потреба и захтева веб апликације, користе се различити типови АПИ.

3.6.1.1. Клијентски или *JavaScript* АПИ

Најједноставнији начин за интеграцију веб апликације са неком од друштвених мрежа је коришћење клијентског односно *JavaScript* АПИ-а. Друштвене мреже обезбеђују *JavaScript* библиотеке које могу да се укључе и користе на било ком веб сајту. Постоје многи додаци друштвених мрежа засноване на овом АПИ-у као што је популарно дугме „свиђа ми се“ (енг. *like*) или „улогуј се“, (енг. *log in*) аутентификација корисника преко неке од друштвених мрежа као производ интеграције базираним на подацима са друштвених мрежа. Овај АПИ користи имплицитни *OAuth* протокол. Сви *JavaScript* АПИ-и користе *JSON* (енг. *JavaScript Object Notation*) [33] формат за репрезентацију података.

Као посебан додатак које одређене друштвене мреже поседују јесте проток података у реалном времена (енг. *streaming*). Овај додатак аутоматски ажурира своје податке уколико постоје нови подаци објављени на корисников профилу (пример: *Twitter Stream Widget*).

3.6.1.2. Серверски или REST АПИ

Серверски АПИ је оријентисан ка серверској изради апликација. Овај АПИ омогућава приступ подацима корисника без њиховог приказа кориснику. Већина друштвених мрежа подржава *REST* (енг. *Representational state transfer*) принципе адресирања, одрживости и репрезентације података. *REST* принципи и *REST* оријентисана архитектура засновани су на *HTTP* (енг. *Hypertext Transfer Protocol*) [34] протоколу, односно његовим методама за приступ, названим *HTTP* глаголима (енг. *HTTP Verbs*) као што су:

- *GET* за читање података,
- *POST* за додавање података,
- *PUT* за измену података,
- *DELETE* за брисање података.

Неки од *REST* АПИ-а одређених друштвених мрежа подржавају различите формате за приказ података, поред *JSON* типа. Најчешће подржани формати су: *XML* (енг. *Extensible Markup Language*) [35] и *XML* проширене верзије као што су: *RSS* (енг. *Rich Site Summary*) [36] и *ATOM* [37]. *REST* АПИ има одређене предности као што је обележавање страна (енг. *pagination*) код великих скупова података. Овај АПИ омогућава код неких друштвених мрежа приступ сваком податку на мрежи преко одређене веб адресе, тако да се тим подацима може приступити преко те адресе, и адресе тих података могу се наћи у другим подацима.

Адресе података добијене *REST* АПИ-ем могу бити различите врсте. Подаци могу бити приказани као једна адреса на којој се налазе сви тражени подаци или могу бити подељени на више адреса и то по типу података. У случају да се подаци налазе на једној адреси, приказ података садржи поље које описује добијену врсту података.

Као што је и очекивано, свака друштвена мрежа има своју структуру података за приказ података једног корисника. Поређењем података једног корисника на две или више друштвених мреже може се добити униформи интерфејс који може помоћи програмерима уколико развијају једну апликацију на више друштвених мрежа.

3.6.2. Поређење апликационих програмских интерфејса друштвених мрежа *Facebook*, *Twitter*, *Google+*, *LinkedIn*

Апликациони програмски интерфејси друштвених мрежа имају међусобне сличности и разлике. У наредним табелама, биће приказане сличности код друштвених мрежа *Facebook*, *Twitter*, *Google+* и *LinkedIn*.

Табела 1. Стандард за ауторизацију – *OAuth*

	<i>Facebook</i>	<i>Twitter</i>	<i>Google+</i>	<i>LinkedIn</i>
Верзија	2.0	1.0a	2.0	2.0* 1.0a
Ауторизациони код	Да	Није подржано стандардом	Да	Није подржано стандардом
Имплицитна ауторизација	Да	Није подржано стандардом	Да	Није подржано стандардом
Власник података	Не	Није подржано стандардом	Не	Није подржано стандардом
Одобравање корисника	Да	Није подржано стандардом	Не	Није подржано стандардом
Опсег привилегија	Да	Није подржано стандардом	Да	Није подржано стандардом

Све друштвене мреже подржавају *OAuth* стандард (табела 2.), али верзије овог стандарда се разликују, па имамо 1.0a (*Twitter*, *LinkedIn*) и 2.0 (*Facebook*, *Google+*). Захтеви за имплементацију могу се пронаћи на форумима друштвених мрежа који се константно ажурирају, па је могуће у кратком времену пронаћи информацију да су неке од друштвених мрежа имплементирале стандард 2.0, уместо тренутног 1.0a.

Међу друштвеним мрежама које подржавају стандард 2.0, подршка којом се добија ауторизациони код је различита. Подржана је серверска ауторизација, тако да веб апликације могу да приступају подацима тих друштвених мрежа уз одређена ограничења. Имплицитна ауторизација примарно се користи код *JavaScript* апликација. Иако *LinkedIn* је делимично подржао стандард 2.0*, њихов *JavaScript* API не подржава овакву врсту ауторизације. Одобравање корисника подржава једино друштвена мрежа *Facebook*. Корисник одобрава веб апликацијама захтев за приступ својим подацима (енг. *apprequests*) који се налазе на друштвеној мрежи *Facebook*, у ограничења приступу подацима. Друштвена мрежа *Facebook* је *OAuth* 2.0 стандард највише имплементирала у својима АПИ-има.

Опсег *OAuth* привилегија различит је на свим друштвеним мрежама. Најчешће су привилегије класификоване ка кориснику и његовим пријатељима. Многе од њих оријентисане су ка деловима профила корисника, друге ка скуповима података као што су слике, видео записи и слично.

На друштвеној мрежи *Facebook* дефинисано је више од 60 различитих типова привилегија као што су: *user_about_me*, *friends_about_me*, *user_activities*, *friends_activities*, *user_birthday*, *friends_birthday* и друге [38]. Друштвена мрежа *Google+* има различите варијанте *OAuth* привилегија [39]. Она омогућава различите скупове привилегија за сваку *Google* апликацију (*Analytics*, *Blogger*, *Calendar*, *Gmail* и друге), и због тога је архитектура ове друштвене мреже више модуларна него друштвене мреже *Facebook*.





Табела 2. Типови АПИ-а

	<i>Facebook</i>	<i>Twitter</i>	<i>Google+</i>	<i>LinkedIn</i>
АПИ				
<i>JavaScript</i>	Да	Да	Да	Да
<i>REST</i>	Да	Да	Да	Да
<i>HTTP Verbs</i>	<i>GET, POST, DELETE</i>	<i>GET, POST</i>	<i>GET</i>	<i>GET, PUT, POST, DELETE</i>

Табела 3: Подржани формати података

	<i>Facebook</i>	<i>Twitter</i>	<i>Google+</i>	<i>LinkedIn</i>
<i>JSON</i>	Да	Да	Да	Да
<i>XML</i>	Не	Да	Не	Да
<i>RSS</i>	Не	Да	Не	Не
<i>ATOM</i>	Да	Да	Не	Не

Табела 4. Додаци на друштвеним мрежама

	<i>Facebook</i>	<i>Twitter</i>	<i>Google+</i>	<i>LinkedIn</i>
Обележавање страна великог скупа података	Да	Да	Да	Да
Дугмад				

Све наведене друштвене мреже поседују додаток, логин дугме, које омогућава веб апликацијама аутентификацију и информације о кориснику са неке од друштвених мрежа. Ово је део *JavaScript* АПИ-а. Популарни додаток наведених друштвених мрежа је дугме „свиђа ми се“, које омогућава кориснику да објави неку новост на свом профилу. Ово дугме има различит назив у код свих наведених друштвених мрежа. Код друштвене мреже *Facebook* то је дугме „like“, *Google+* (+1), *Twitter* (tweet), *LinkedIn* (share) (табела 3).

Друштвена мрежа *Facebook* има више доступних додатака у односу на друге друштвене мреже и они се веома лако интегришу у веб презентацијама. То су: *Send Button*, *Embedded Posts*, *Follow Button*, *Comments*, *Share Dialog*, *Activity Feed*, *Recommendations Box*, *Recommendations Bar*, *Like Box*, *Registration*, *Facepile* [38].

REST АПИ друштвених мрежа подржан *HTTP* протоколом. Овај АПИ прате различите привилегије за приступ подацима друштвених мрежа. *Google+* друштвена

мрежа подржава само операцију за читање података – *GET*. Друштвена мрежа *Twitter* подржава *GET* и *POST* – операције за читање и додавање података, док је код ове друштвене мреже постоји и подршка за измену и брисање података али је то имплементирано кроз *POST* операцију. Друштвена мрежа *Facebook* има операције за читања, додавања и брисање података (*DELETE*), док је подршка за измену података имплементирана кроз операцију *POST*. Друштвена мрежа *LinkedIn* је најкомплетније имплементирала стандард *HTTP*, где је за свака операција одвојена једна од друге. Постоје операције за читање, додавање, измену и брисање података. (табела 2)

Формат података који је подржавају све наведене друштвене мреже јесте *JSON*. Овај формат података се веома лако користи са *JavaScript* АПИ-ем. *XML* формат користе *Twitter* и *LinkedIn*. Међутим, постоје још 2 специфична формата *XML* која подржава *Twitter* друштвена мрежа и то су: *RSS* и *ATOM*. *ATOM* формат је делимично подржан код протока података у реалном времену, код *Facebook*-а.

Код обележавања страна великог скупа података, свака друштвена мрежа на различит начин користи параметре за обележавање страна (*limit*, *offset*, *page*, *count*, *per_page*).

Најчешће *REST* АПИ обезбеђују податке на посебним адресама. Друштвена мрежа *Facebook* је једина мрежа која омогућава приказ података на једној адреси и то је могуће кроз такозвани *Graph* АПИ [38]. *Graph* АПИ је направљен као један интерфејс ка свим објектима који се налазе на мрежи. Ова друштвена мрежа користи *Opengraph* protocol [38] за додавање мета података на *HTML* странама, користећи мета тагове у *HTML* head делу стране. Интересантна предност *Opengraph* је та да друштвена мрежа користи мета податке да би се повезала са *Facebook* апликацијом. Пример ових тагова јесу специјалним таговима *fb:admins* и *fb:app_id*, који се користе за добијање администраторских дозвола у апликацији.

Табела 5. Структура података у *JSON* формату једног корисника

<i>Facebook</i>	<i>Twitter</i>	<i>Google+</i>	<i>LinkedIn</i>
<i>username</i>	<i>displayName</i>	<i>name</i>	<i>first-name</i>
<i>name</i>	<i>name(object)</i>	<i>screen_name</i>	<i>last-name</i>
<i>first_name</i>	<i>nickname</i>		
<i>middle_name</i>			
<i>last_name</i>			

Све друштвене мреже имају дефинисану структуру података за одређеног корисника. Сличност за *JSON* формат података код свих друштвених мрежа за одређеног корисника је атрибут *id* који јединствено одређује једног корисника на некој од друштвених мрежа. Сви други атрибути корисника се разликују на свим друштвеним мрежама.

3.6.3. Апликациони програмски интерфејси друштвене мреже *Facebook*

На друштвеној мрежи *Facebook* постоји неколико типова АПИ-а који могу да се користе при развоју апликација које се интегришу на овој друштвеној мрежи. Три најбитнија АПИ-а за ову друштвену мрежу јесу:

- iOS АПИ [38]
- Android АПИ [38]
- Web АПИ [38]

Прва два горе наведена АПИ-а користе се за израду апликација на мобилним платформама за оперативне системе iOS [40] и Android [41]. Међутим, рад се бави развојем апликације засноване на *Facebook Web* АПИ-у. У наставку рада биће занемарени описи прва два наведена АПИ-а, и детаљније ће бити приказан *Web* АПИ.

Web АПИ подржава више програмских алата (енг. *Software Development Kit*) за различите програмске језике који могу да се користе за развој апликација. Подржани су:

- *JavaScript*
- *PHP*
- *Flash*
- *Python*
- *Java (Swing), Spring Social*
- *Java (Blackberry) by RIM*
- *C#*
- *Ruby*
- *Node.js*

Због веома великог броја подржаних програмских језика, биће описана два програмска језика за развој апликације и то *JavaScript*, односно клијентски АПИ за развој апликација и *PHP* [42], односно серверски АПИ за развој апликација.

3.6.3.1. *JavaScript* АПИ друштвене мреже *Facebook*

Facebook SDK JavaScript библиотека је садржи богат скуп функција којима је омогућена доступност корисничким подацима на овој друштвеној мрежи. Да би се користио *JavaScript* АПИ апликација мора бити регистрована, чиме се добија јединствен апликациони идентификатор. Учитавање и иницијализација овог АПИ-а могућа је уколико је валидан јединствен апликациони идентификатор [38].

Постоје више група метода које се налазе у овој библиотеци. То су:

- Основне методе (енг. *Core Methods*)
- Методе за аутентификацију (енг. *Auth Methods*)
- Методе за обраду догађаја (енг. *Event Handling*)
- Методе за обраду XFBML (енг. *XFBML*)
- *Canvas* методе (енг. *Canvas Methods*)

У зависности од захтева апликације, користе се различите групе метода. Основне методе користе се у апликацијама, зато што њима прво иницијализује АПИ библиотека.

У основним методама овог АПИ-а постоје три методе и то:

1. *FB.api* – метода којом може да се приступа *Graph* АПИ-у [38].
2. *FB.init* – метода којом се иницијализује АПИ библиотека

3. *FB.ui* – метода којом се отварају дијалози друштвене мреже као искачући прозори, модални дијалози или целе стране. [38]

У методама за аутентификацију постоје четири методе и то:

1. *FB.getAuthResponse* – метода за синхрони приступ тренутном ауторизацијом статусу
2. *FB.getLoginStatus* – метода за проверу статуса на серверу и добијања сесије ако је корисник повезан са апликацијом
3. *FB.login* – метода за пријаву, за ауторизација и за привилегије
4. *FB.logout* – метода за одјаву корисника

Основним методама и методама за аутентификацију могуће је направити апликацију на овој друштвеној мрежи користећи овај АПИ. Ове две групе метода можемо назвати неопходним скупом метода за развој апликација коришћењем *Javascript* АПИ-а .

3.6.3.2. PHP АПИ друштвене мреже Facebook

Facebook SDK PHP библиотека омогућава богат скуп серверских функција за приступ *Facebook*-овом серверском АПИ-у (*REST* АПИ-у). Ова библиотека садржи скуп функција којим се приступа *Graph* АПИ-у и скуп функција којим могу да се пишу упити ка *Graph* АПИ-у.

Ова библиотека се најчешће користи за развој административног дела апликације, али могуће је делимично користити и за тренутну сесију корисника. Уколико је потребно брисање ауторизационог кода (енг. *access token*), ова библиотека може веома просто да обави ту функцију уз поновно покретање процеса аутентификације и ауторизације апликације код корисника чији је код избрисан.

Као код *Javascript* АПИ-а, тако и код *PHP* АПИ-а потребно иницијализовати библиотеку [43], са јединственим апликационим идентификатором и са апликационим сигурносним кључем (енг. *App Secret*) [38]. Ова библиотека може да се користи на веб сајту за аутентификацију корисника са *Facebook* налогом. На серверу, библиотека има у себи функцију за аутентификацију, функцију за захтевање привилегија и функцију за одјаву корисника. То су функције: *Facebook::getUser()*, *Facebook::getLoginUrl()* и *Facebook::getLogoutUrl()*. Могуће је интегрисати обе библиотеке у једној апликацији.

3.7. Дефинисање друштвене мреже за развој апликације

Тема овог рада могла би се дефинисати као израда апликације за тестирање математичког знања на платформама неких од актуелних друштвених мрежа. Платформа би означавала домен, односно коришћење АПИ-а неке друштвене мреже за израду саме апликације. Анализирајући тренутно стање актуелних друштвених мрежа и на основу броја корисника који су активни на актуелним друштвеним мрежама у Србији, као домен односно платформу за израду апликације за тестирање

математичког знања одабрана је друштвена мрежа *Facebook*, као најактуелнија друштвена мрежа у Србији.

4. Методологије за развој софтвера

Методологије за развој софтвера веома су битне у самом процесу развоја софтвера било да се захтеви за развој софтвера креирају по некој од познатих методологија или не. Међутим, у данашње време методологије за развој софтвера се мењају. Некадашње методологије не задовољавају данашње критеријуме развоја [44]. Главни разлог због ког се мењају методологије јесте електронско и мобилно пословање које све више преовладава тржиштем. Старе технике за развој софтвера називају се "тешким" методологијама. Ови типови методологија се тешко уклапају у развој пројеката код електронског и мобилног пословања. Због тога данас све више имамо "лакше" методологије за развој софтвера које имају више практичних решења од тзв. "тешких" методологија. Не постоји савршена методологија за развој софтвера. Свака има своје предности и мане. Методологију за развој софтвера бирају менаџери пројекта и од њих зависи која ће технологија да се примени за развој неког софтвера. Методологија и стратегија за развој неког софтвера зависи од више параметара као што су:

- буџет,
- величина тима који реализује пројекат,
- технологија која се користи,
- документација,
- обука,
- најбоље применљиве праксе,
- алати и технике,
- постојећи процеси,
- и слично.

У наставку рада биће обрађене следеће методологије:

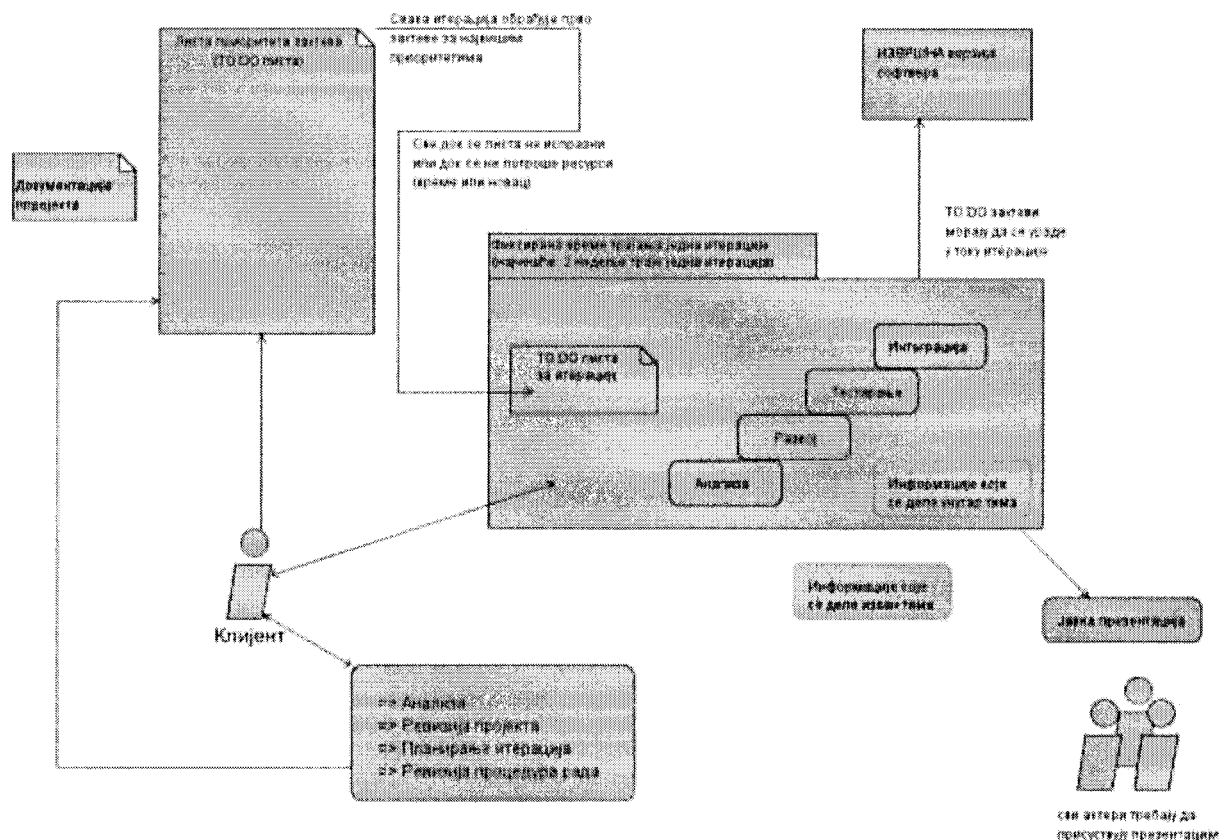
- агилна (енг. *Agile*) методологија развоја софтвера [45],
- скрам (енг. *Scrum*) методологија развоја софтвера [46],
- обједињени развојни процес софтвера (енг. *RUP*) [47].

Циљ обраде и анализе јесте добијање најпогодније методологије за развој апликације у домену друштвених мрежа, односно друштвене мреже *Facebook*.

4.1. Агилна методологија развоја софтвера

Агилна методологија развоја софтвера је група метода за развој софтвера базирана на итеративном и инкременталном развоју где захтеви и решења еволуирају путем сарадње унутар тима [45]. Тим најчешће чине људи који су експерти у различитим областима, а који раде на остварењу заједничког циља [48]. Агилна метода укључује константну комуникацију у току развоја и ако је могуће одржавање састанака након написане документације. Многи тимови се налазе у веома малом

простору, односно најчешће у једној канцеларији су сви чланови тима неопходни за развој софтвера.



Слика 9. Дијаграм агилне методологије развоја софтвера

Тим чине програмери и људи који дефинишу производ, односно менаџери производа, бизнис аналитичари и стварни купци производа. Такође, потребно је укључити још и тестере (тест програмере), дизајнере, људе задужене за техничку документацију и менаџмент.

Ова методологија обухвата прилагодљиво планирање, еволутивни развој и испоруку пројекта, временски ограничен итеративни приступ и захтева брзе и флексибилне промене. Агилна методологија може да се дефинише и као систем метода дизајнираних тако да минимизују цену промена, нарочито у контексту када је време реализације пројекта битно. Ова методологија примењује се још у ситуацијама где је потребно прилагодити развој важним факторима које није могуће лако контролисати или где је потребан веома брз развој софтвера.

4.2. Scrum методологија развоја софтвера

Scrum представља агилни приступ за управљање развојем софтвера [46]. Јавља се половином 90 - их година прошлог века. Scrum није процес или техника за стварање производа, он је више оквир у којем је могуће применити различите процесе и технике. Scrum разјашњава релативну ефикасност производног управљања и

развојних пракси како би се остварио напредак [49]. Термин “*scrum*” потиче из стратегије у игри рагбију где означава моменат када се противнички тимови скупљају око лопте (“добијање лопте која је била у ауту (изван игре) и враћања лопте у игру”) и боре за посед лопте. Веза са софтверским пројектима је само симболичка.

Scrum представља итеративни, инкрементални агилни приступ за развој софтвера. Концентрисан је на то како чланови тима треба да функционишу у циљу да се произведе флексибилан систем при константној промени окружења. На крају сваке итерације добија се потенцијални скуп функционалности [50].

Ова метода је више везана за агилно управљање софтверским пројектом него за агилно пројектовање софтвера. Она прописује начине управљања захтевима, формирање итерација, контроле имплементације и испоруке клијенту.

Оквир скрама се састоји из *scrum* тимова и њихових улога, догађаја, артефаката и правила. Сваки елемент садржан у оквиру скрама игра посебну улогу и суштински је за успех и његово коришћење.

Основу представљају три кључна питања која се постављају на свакодневним, јутарњим “стојећим” састанцима који трају 15 минута:

- 1 Шта је урађено јуче?
- 2 Шта ће се радити данас?
- 3 Какве нас данас препреке очекују?

Ова питања се односе на:

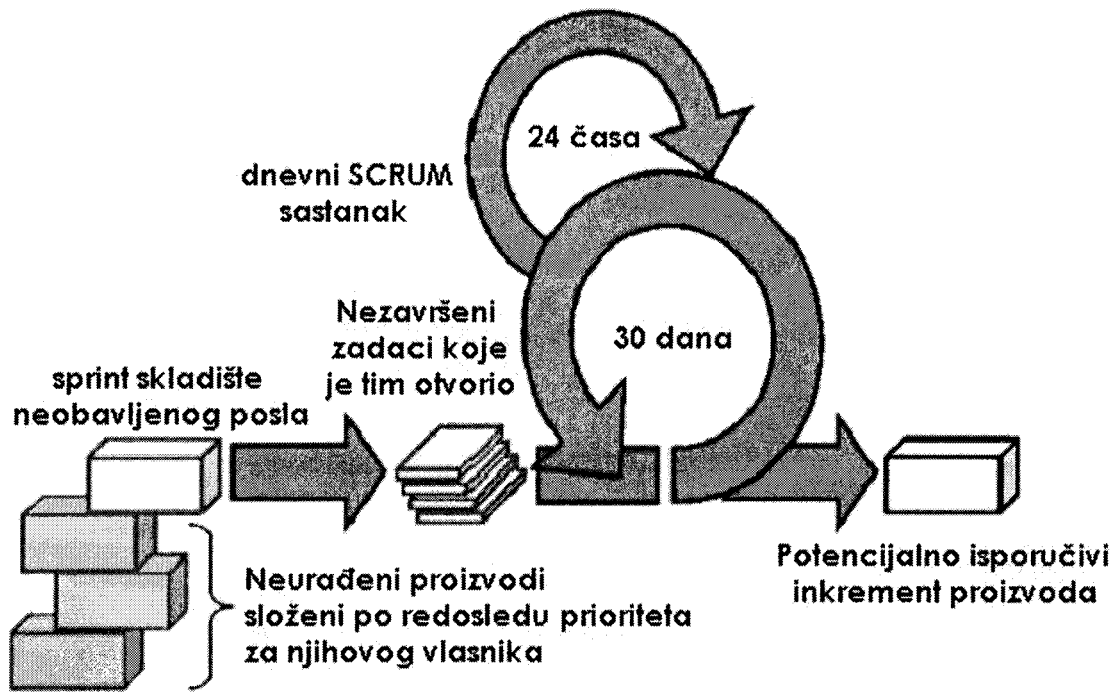
- 1 контролу извршеног,
- 2 планирање будућег дизајна,
- 3 идентификацију ризика и налажење решења.

Scrum се заснива на бази емпиријске процесне контролне теорије, тј. емпиризма. Емпиризам тврди да знање произилази из искуства и доношења одлука на основу онога што је научено. *Scrum* примењује учестан, инкременталан приступ како би оптимизовао предвидљивост и контролисао ризик. Имплементацију емпиријског процеса подржавају три стуба: транспарентност, инспекција и адаптација.

Скрам прописује четири формалне прилике за инспекцију и адаптацију.

- 1 Састанак и планирања *sprint*
- 2 Дневни *scrum*
- 3 Преглед *sprint*
- 4 Ретроспектива *sprint*

Sprint је срце *Scrum* методологије. Он представља временски оквир од месец дана или краћег временског периода у оквиру којег се „Обављени“, корисни, и потенцијално дистрибутивни инкремент производа ствара. *Sprint*-еви имају једнаку дужину трајања кроз развојни напор. Нови *sprint* почиње одмах након завршетка претходног *sprint*-а [49]. За то време се свакодневно прати напредак и идентификују спорна и ризична места напредовања



Слика 10. Дијаграм *Scrum* методологије развоја софтвера

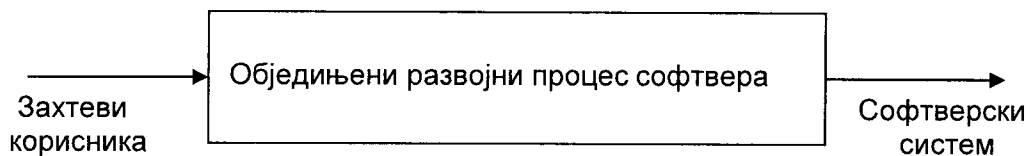
4.3. Обједињени развојни процес софтвера

Обједињени развојни процес софтвера (енг. *Rational Unified Process, RUP*) је резултат добијен из више од 30 година искуства и представља скуп активности које су потребне за трансформацију корисничких захтева у софтверски систем. Обједињени развојни процес је више од једног процеса - то је генерички оквир који може бити специјализован за врло велике класе софтверских система, за различита подручја апликација, различите тимове организације, различите нивое надлежности и пројекте различитих димензија. Заснован је на компонентама што значи да је софтверски систем изграђен од софтверских компоненти које су између себе повезане добро дефинисаним интерфејсом. Разликује се од осталих методологија по томе што није само препорука о начину реализације софтверског пројекта већ је уједно и комплетан производ компаније *Rational Software* која се налази по окриљем *IBM*. Овакав стандардизован приступ може бити атрактиван за велике организације које захтевају заједнички језик и алате који се морају користити у целој компанији као и у интерној и екстерној комуникацији компаније.

Обједињени развојни процес софтвера је метода базирана на итеративно инкременталном моделу животног циклуса софтвера, која користи стратегије управљања према случајевима коришћења (енг. *use-case driven*) и оријентације ка архитектури (енг. *architecture centric*), као и објектно оријентисану методу пројектовања софтвера. За нотацију се користе *UML* (енг. *Unified Modeling Language* [51]) дијаграми.

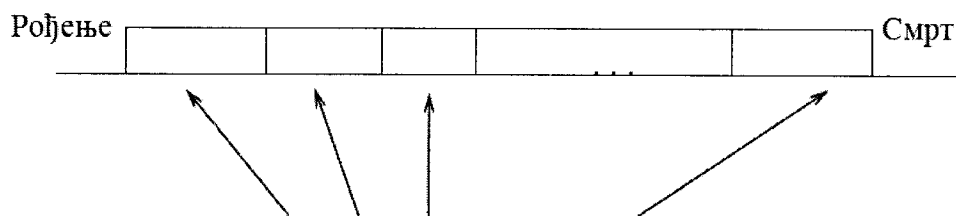
Итеративни развој предлаже прилагодљиво планирање засновано на 2 врсте планова: фазни план и серија итеративних планова. Фазни план је један а итеративних има много више. У сваком тренутку дефинисана су 2 итеративна плана: за текућу и

наредну итерацију. Како се кориснички захтеви и архитектура стабилизују, процењује брзина развоја и тим сазрева, могуће је планирати 2 или 3 итерације унапред.



Слика 11. Обједињени развојни процес софтвера

Обједињени развојни процес се понавља кроз серију циклуса чинећи тако животни циклус система. Сваки циклус се завршава релативно комплетним и постојаним скупом опипљивих информација које су креирали, мењали или користили чланови тима приликом извршавања активности а који је користан за корисника (*release*).



Циклус се завршава извршном верзијом софтвера

Слика 12. Животни циклус процеса састоји се од циклуса од свог рођења па до смрти

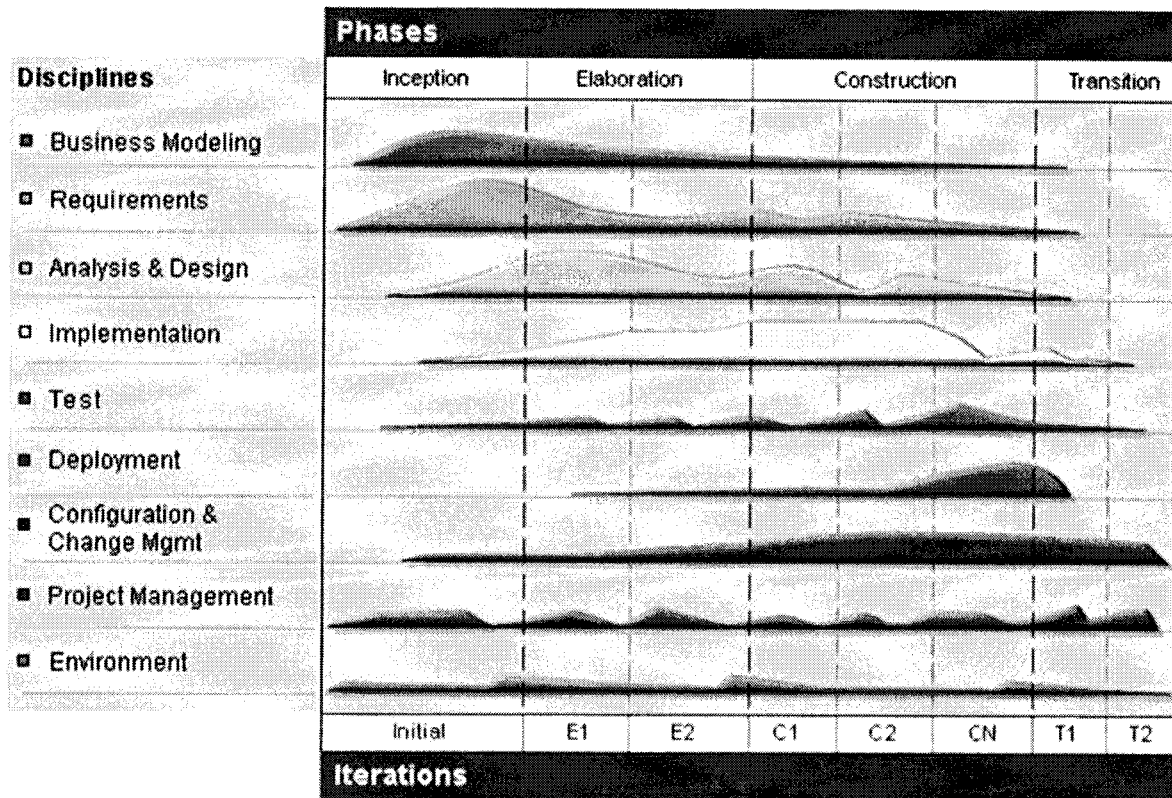
Неке од кључних пракси и смерница јединственог процеса развоја софтвера илуструју његову суштину:

- развој у кратким, временски ограниченим (енг. *timeboxed*) итерацијама,
- развој високо ризичних и елемената са високом вредношћу (на пример, основна архитектура) у раним итерацијама, преферирајући поновно коришћење (*re-use*) постојећих компоненти,
- сигурност и поверење код клијента добија се издавањем неколико бета верзија које му се испоручују,
- прилагођавање променама рано у пројекту,
- усредсређеност чланова тима на тимски рад.

Обједињени развојни процес софтвера се састоји из четири фазе: почетак (*inception*), развој – усавршавање (*elaboration*), грађење (*construction*) и прелаз (*transition*).

- Почетна фаза дефинише опсег пројекта и визију крајњег производа. Овде се наводе основни случајеви-коришћења (СК).
- У току развојне фазе прави се план пројекта, СК се разрађују и даје се нацрт архитектуре система.

- У фази грађења добија се комплетан софтвер који се придружује до архитектуре система. На тај начин се долази до Бета Извршне Верзије (БИВ) (енг. *beta release*) [52].
- У прелазној фази БИВ се прослеђује до корисника ради тестирања. Након исправке уочених проблема прави се генерална извршна верзија (енг. *general release*). У прелазној фази се обучавају корисници.



Слика 13. Девет дисциплина у обједињеном развојном процесу софтвера [53]

Обједињени развојни процес захтева прилагођавање сваком пројекту тј. одабира се скуп радних производа који ће се креирати из великог али опционог скупа који је доступан. Ово јединствено прилагођавање се зове "Случај развоја" (*development case*) пројекта. У општем случају "мање је боље" је водилца и случај развоја би требало да садржи минимум радних производа потребних да би се обрадили ризици и циљеви пројекта.

4.4. Поређење методологија за развој софтвера

Приликом избора методе за развој софтвера, потребно је узети у обзир и искористити нека дефинисана правила:

- *Glass*-ово правило (енг. *Glass law* [54]): Недостаци код дефинисања захтева су основни извор неуспеха у развоју пројекта.
- *Boehm*-ово прво правило (енг. *Boehm's first law* [55]): Грешке се најчешће јављају током активности везаних за корисничке захтеве и пројектовање и ове грешке су све скупље што се касније отклоне.

- *Boehm*-ово друго правило (енг. *Boehm's second law* [56]): Израда прототипова (значајно) смањује грешке приликом дефинисања корисничких захтева и пројектовања, нарочито код корисничког интерфејса.

Агилни софтверски развој подразумева брз и развој са малим степеном формализма. Све агилне методе стављају акценат на итеративно стварање софтвера у кратким временским интервалима. Агилне методе се разликују од итеративних по томе што се тај период код њих мери недељама а не месецима рада, док се рад одвија у духу колективне сарадње. И агилне методе се међу собом могу разликовати по томе да ли се стриктно придржавају предвиђених временских оквира или не.

Методе итеративног развоја софтвера су, са друге стране, ближе методама водопада (енг. *waterfall*) за развој софтвера. Недостатак модуларног тестирања и превелики ниво документације у односу на агилни приступ поставља границе међу овим методологијама. Управљање променама није адекватно решено те промене у каснијим фазама увек доносе велике трошкове и велики ризик. *RUP* суштински зависи од примене UML дијаграма што не мора да буде значајно за агилне методе. *RUP* се стално мења и унапређује уз прихватање агилних принципа.

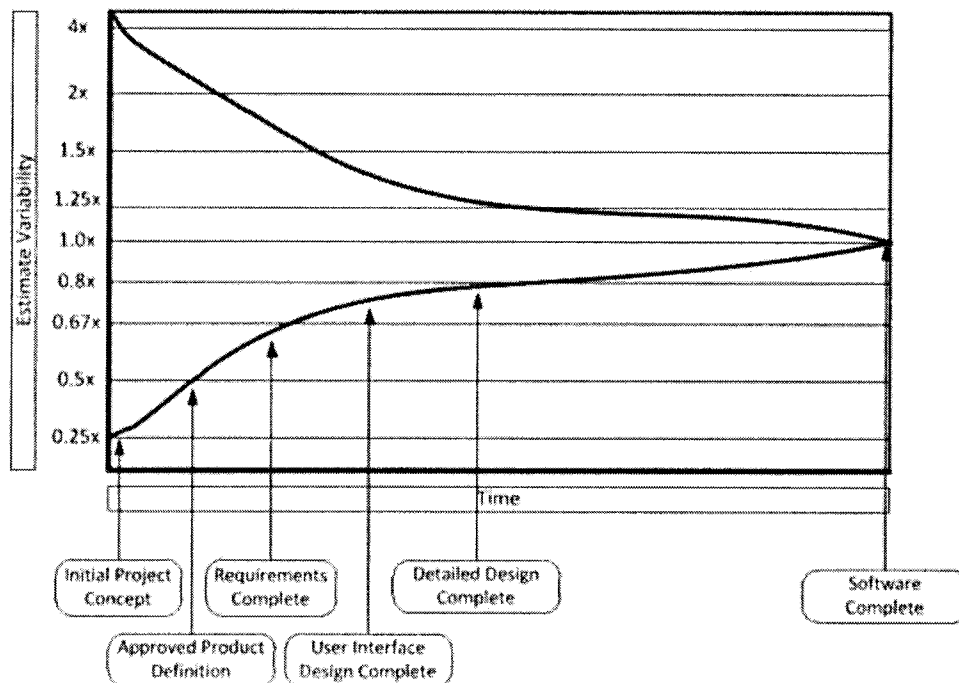
Beck-Fowlerova хипотеза (енг. *Beck-Fowler hypothesis* [57]) објашњава приступ агилних метода и гласи: Методе агилног програмирања смањују утицај промене корисничких захтева.

Хипотеза може да се протумачи тако да агилне методологије имају велики значај код развоја софтверски система код којих кориснички захтеви не могу да се спецификују унапред.

Неки од фактора који спречавају поуздано креирање спецификације унапред су:

- Клијенти или корисници нису сигурни шта желе
- Они имају проблема да објасне све што желе и знају
- Многи детаљи онога што они желе ће бити откривени тек током развоја
- Детаљи су превише комплексни за људе
- Они мењају мишљења док се производ развија
- Спољне силе (као што су конкурентски производ или услуга) воде ка изменама или захтевима за промене унутар корисничких захтева

Утицај методе за развој на процене у развоју софтвера описан је као: „Итеративни пројекти брже прелазе до усавршења њихових процена коришћењем података који су специфични за пројекат” [58]. Због раних промена у захтевима и других фактора, постоји иницијална доза велике несигурности, која опада како време пролази и прикупљају се информације. Овај термин је познат као купа несигурности (енг. *cone of uncertainty*) и приказан је на слици 14.



Слика 14. Купа несигурности код агилних метода развоја софтвера [59]

Купа несигурности дефинише статистички предвидиве нивое неизвесности у пројекту и приказује сваки ниво неизвесности током пројекта. То је један од начина графичког приказа зависности више параметара у односу на време израде софтверског пројекта.

Методологија за развој софтвера описује процес који користе инжењери и програмери у креирању и развоју софтверског производа. Особине методологије описују захтеве, контролу развоја, обуку и одржавање система. Кад год се користи методологија за развој софтвера, циљ њеног коришћења јесте стварање система који задовољава примарни циљ за развој софтвера, и то у дефинисаном року и са дефинисаним трошковима. Методологија такође укључује и увођење и одржавање софтвера који треба да има карактеристике једноставне употребе, инсталације и минимизовање грешака током развоја. Због особина методологија и великог броја фактора који могу утицати на сам избор потребно је направити поређење како би постојао увид у методологије и њене предности и мане.

У табели 6. приказано је поређење *RUP* и *Scrum* методологија [60].

Табела 6. Поређење *RUP* и *Scrum* методологија

Број	Особина	<i>RUP</i>	<i>Scrum</i>
1	Спецификација свих захтева од почетка пројекта	Не	Не свих и често се мењају
2	Трошкови пројекта	Најчешће као процењена	Најчешће као процењена
3	Гаранција успеха	Веома висока	Висока

4	Потребно знање	Високо	Веома високо
5	Преклапање фаза	Да	Да
6	Сложеност	Веома сложен процес	Сложен процес
7	Тип радне платформе	Еволутивни, интегрисани, итеративни	Итеративни и инкрементални
8	Трошкови преправке	Високи	Високи
9	Тестирање	У свим фазама	После фазе кодирања
10	Учешће клијента	Високо, после сваке итерације	Високо
11	Потреба за познавањем основног процеса пословања	Потребно	Веома добро познавање
12	Погодна величина пројекта	Опсег великих пројеката	Опсег малих и средњих пројеката
13	Контрола трошкова	Не	Не
14	Једноставност	Комплексан	Прост
15	Укључени ризик	Низак	Није висок
16	Флексибилност	Веома флексибилан	Флексибилан
17	Укључење промене	Лако	Лако
18	Поновна употребљивост	Да	Да
19	Документација и обука	Веома ограничено	Ограничено
20	Временски оквир	Дуг	Умерен
21	Доступност радног софтвера	Зависи од модела развоја	На крају сваке итерације
22	Прилагођавање производа	Могуће	Могуће
23	Контрола клијента преко администратора	Да	Да
24	Потребна креативност у тиму	Да	Да
25	Размена знања	Да	Да
26	Величина тима	Велики тим	Велики тим
27	Примарни циљ	Висок квалитет	Брз развој
28	Имплементација	Комплексна	Умерена

II. Развој апликације за тестирање математичког знања у домену друштвене мреже *Facebook*

Развојни процес софтвера је метод за организацију активности које се односе на креирање и одржавање софтверског система. Овај пројекат приказује развојни процес који описује могући редослед активности у развојном циклусу. Дакле, не дефинише се стандардан процес већ се стандардизују активности у развојном циклусу. Кораци (активности) у развојном циклусу зависе од природе процеса која је условљена природом проблема. Развојни процес је итеративан, односно састоји се од вишеструких развојних "кругова" прикупљања корисничких захтева, анализе система, пројектовања, имплементације и тестирања. У односу на горе наведе методологије овде ће се приказати пројектна документација која не одговара стриктно једној методологији већ представља комбинацију методологија које се примењују у развоју апликације. С обзиром да се рад не ради у тиму, одређене методе из методологија биће занемарене.

Потребно је пројектовати систем (апликацију) који омогућава решавање тестова у домену друштвених мрежа. Учесници (актери) су корисници истих и администратор система.

Основне активности учесника:

Корисник: Приступ апликацији, Тестирање (даје одговоре), Креирање квизова (уколико је корисник професор математике) , Позива пријатеље да тестирају своје знање, Одјављује резултате теста на свом профилу

Администратор: Одржавање апликације, одобравање приступа корисничком окружењу за креирање квизова

Приступ апликацији може имати свако ко је регистровани корисник друштвене мреже *Facebook*. Сваки конкретан тест чува се у бази података. Одговори су по тежини класификовани у три групе: лаки, средњи и тешки. Сваки одговор носи исти број поена. Конкретан тест треба да има око 25% лаких, 50% средњих и 25% тешких питања. Титуле се достављају кориснику након завршеног текста са укупним бројем поена.

Ако је корисник професор математике, одобравање приступа корисничком окружењу за креирање квизова могуће је уколико је корисник послао захтев за приступ овом окружењу.

Само полагање теста се састоји у томе да корисник даје одговоре на постављена питања. Питања већ садрже понуђене одговоре. Један од понуђених одговора је тачан. Ово је пример затвореног теста. Када заврши тест, кориснику је омогућен увид у резултат теста. Поред тога, постоји могућност да се ови резултати прикажу на профилу корисника, уколико корисник објави резултате на профилу.

Алати који се користе у развоју пројектне документације и имплементацији су:

- За дефинисање пројектне документације користи се алат *Visual Paradigm* [61].
- За развој пројекта користи се развојно окружење *NetBeans* [62].
- Пројекат се развија као веб апликација скуп програма за развој - *XAMPP* (*Apache, MySQL, PHP*) [63].

5. Прикупљање и дефинисање захтева

5.1 Текстуална анализа захтева

Математички квиз у домену друштвених мрежа

1. Математички квиз у домену фејсбук друштвене мреже(ФБ)

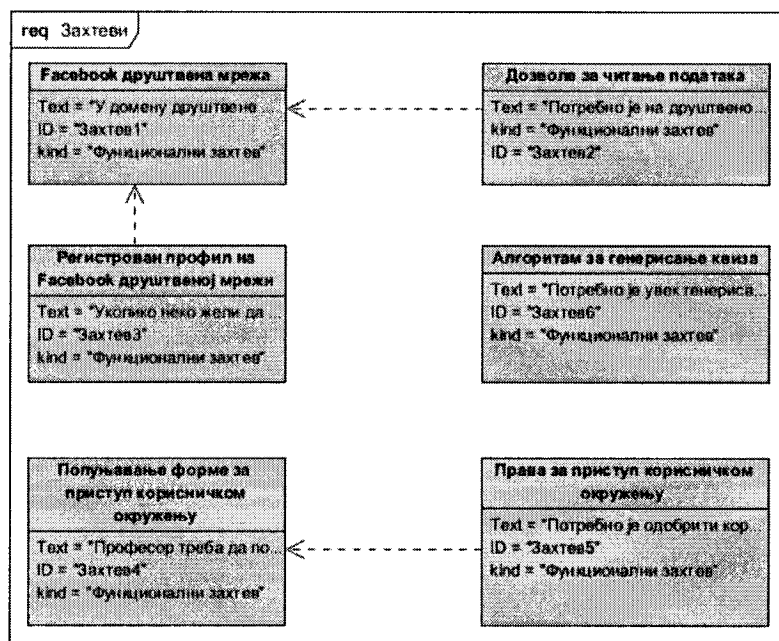
Реализација математичког квиза у домену фејсбук друштвене мреже замишљена је да функционише на следећи начин:

- приказ почетне странице математичког квиза (дугме за дозвољавање читања личних података корисник-а квиза);
- након „уласка“ (дугме улаз) у апликацију (одобравања и читање података на друштвеној мрежи) корисник-у се нуде различити квизови, на основу његових година (прва предност ФБ друштвене мреже);
- након одабира квиза, на основу алгоритма, који генерише квиз, корисник добија квиз са питањима различите тежине и то: 25% тешких питања, 25% лаких питања и 50% средњих питања;
- на основу одговора на добијена питања корисник добија различиту титулу, након завршетка квиза;
- титулу корисник може да подели са својим пријатељима (друга предност ФБ друштвене мреже);
- после завршетка квиза корисник може да позове своје пријатеље да тестирају своје математичко знање у апликацији (трећа предност ФБ друштвене мреже).
- корисник има могућност генерисања квизова на основу критеријума, и то на основу разреда и математичке области
- професори математике могу да креирају квизове за своје ученике.
- да би професор математике добио права за приступ корисничком окружењу, потребно је прво да попуни кратку форму са информацијама: где предаје математику (у којој школи ради) и у ком граду ради.
- професор предлаже ученицима квиз са задацима. Услов за решавање овог квиза јесте: ученици морају да имају регистрован профил на фејсбук друштвеној мрежи.

Текстуална анализа користи се за лакше пројектовање софтвера и издвајање елемената који ће се користити у пројектној документацији.

- Плавом бојом (■) обележени су захтеви које је потребно реализовати у пројекту.
- Наранџастом бојом (■) обележене су улоге које се појављују у пројекту.
- Зеленом бојом (■) обележени су случајеви коришћења који се појављују у пројекту.

5.2 Дијаграм захтева



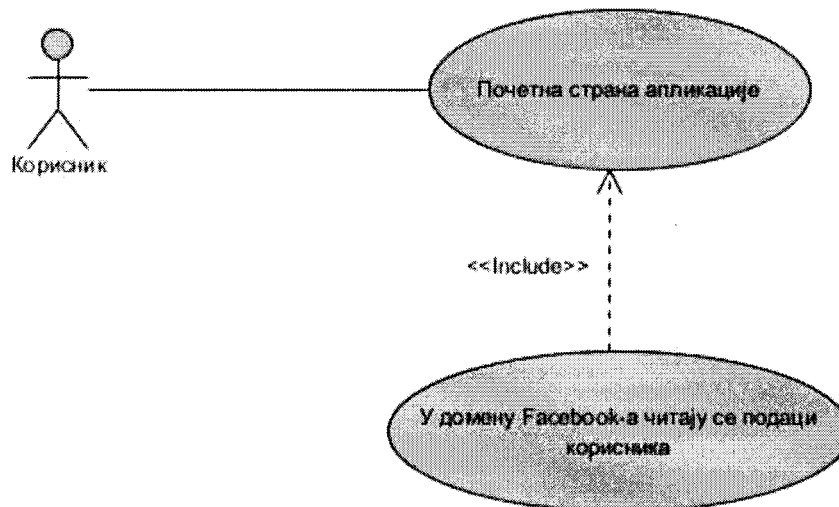
Слика 15. Дијаграм захтева

Табела 2. Текстуална документација дијаграма захтева

Facebook друштвена мрежа	У домену друштвене мреже <i>Facebook</i> потребно је направити апликацију која омогућава корисницима друштвене мреже да тестирају своје знање из математике
Дозволе за читање података	Потребно је на друштвеној мрежи прочитати одређене податке корисника.
Регистрован профил на Facebook друштвеној мрежи	Уколико неко жели да решава било који квиз, потребно је да поседује валидан регистрован профил на друштвеној мрежи Facebook
Попуњавање форме за приступ корисничком окружењу	Професор треба да попуни форму са информацијама за приступ корисничком окружењу за креирање квицова
Права за приступ корисничком окружењу	Потребно је одобрити кориснику апликације коришћење корисничког окружења за креирање квицова
Алгоритам за генерисање квица	Потребно је увек генерисати јединствени квиз за корисника

5.3. Дијаграми случајева употребе

5.3.1. Приказ почетне стране апликације

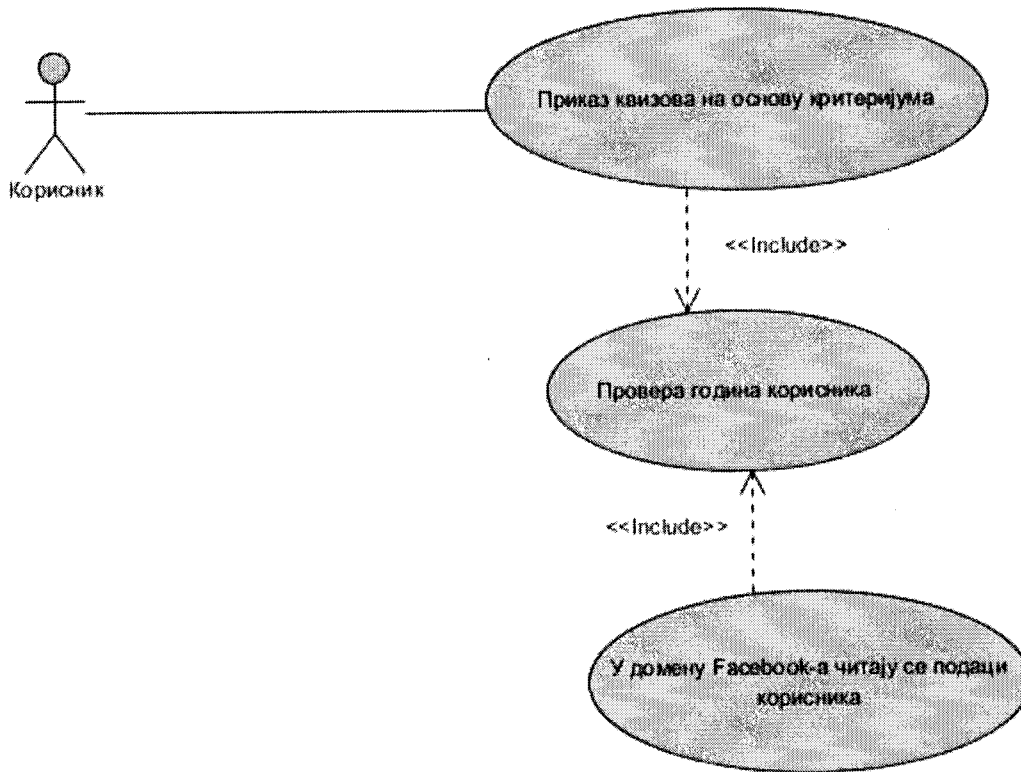


Слика 16: Дијаграм случаја употребе

Опис

ИД	НАЗИВ	ОПИС
1	Корисник	Корисник је свако ко има регистрован налог на друштвеној мрежи <i>Facebook</i>
2	Почетна страна апликације	Доласком на адресу апликације (https://apps.facebook.com/matematickikviz/) отвара се почетна страна апликације. На почетној страни, корисник може да прочита основне информације о апликацији и уколико жели да тестира своје знање из математике, потребно је да одобри апликацији да чита његове податке на друштвеној мрежи <i>Facebook</i>
3	У домену <i>Facebook</i> -а читају се подаци корисника	Уколико корисник одобри апликацији да чита његове податке, онда се ти подаци корисника читају из базе података друштвене мреже <i>Facebook</i> на основу захтева који је послала апликација.

5.3.2. Приказ генерисаних квизова

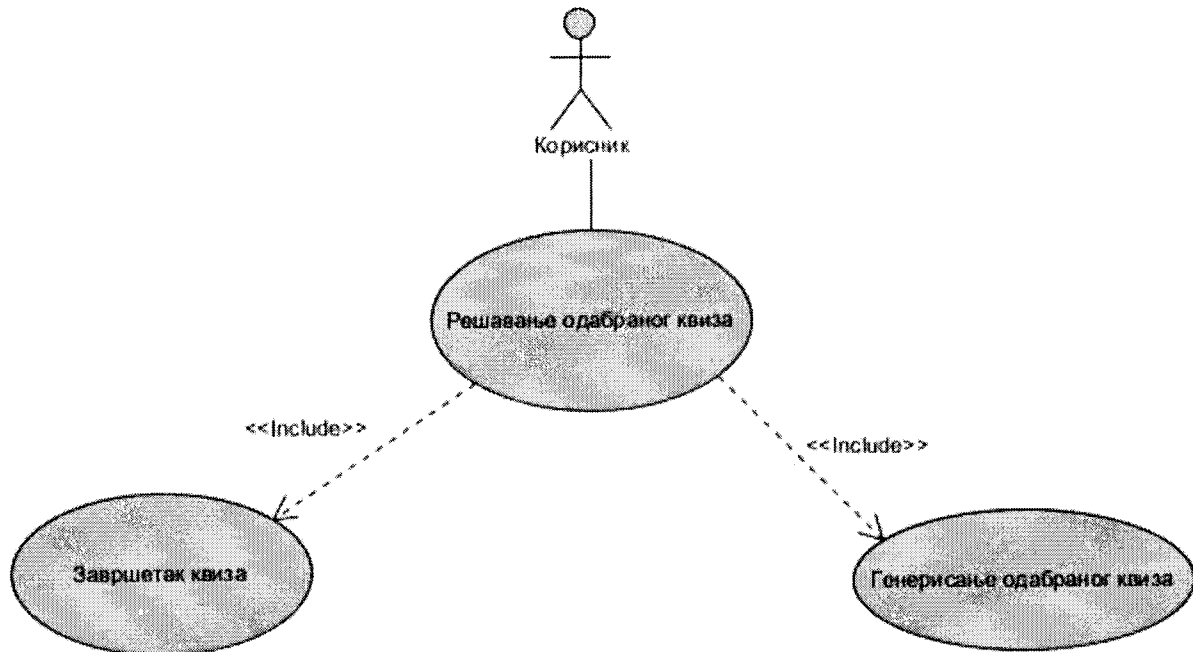


Слика 17: Дијаграм случаја употребе

Опис

Учесник	Документација
Корисник	Корисник је свако ко има регистрован налог на друштвеној мрежи <i>Facebook</i>
Приказ квизова на основу критеријума	Приказују се кориснику квизови на основу одређеног критеријума. Један могући критеријум јесу године корисника на основу којих може да се врши предвиђање знања корисника из математике.
Провера година корисника	Критеријум за генерисање квизова за корисника

5.3.3. Решавање квиза

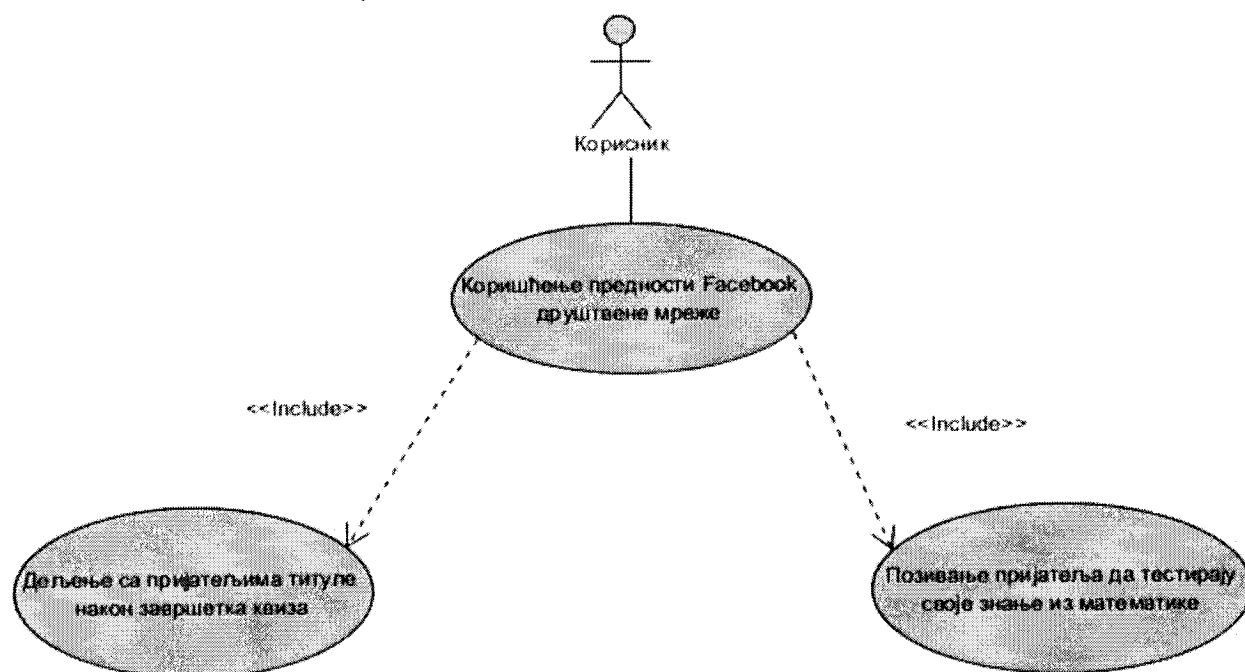


Слика 18: Дијаграм случаја употребе

Опис

Назив	Документација
Корисник	Корисник је свако ко има регистрован налог на друштвеној мрежи <i>Facebook</i>
Генерисање одабраног квиза	Кориснику се приказује одређени број квизова на основу критеријума. Постоји могућност да корисник изабере један од понуђених квизова. Након изабарања једног од квизова, за дати квиз, генеришу се питања која су у релацији са изабраним квизом
Решавање одабраног квиза	Кориснику се приказују питања за дати квиз. Корисник може да изабере један од понуђених одговора за дато питање. Након „закруживања“ свих одговора, потребно је завршити дати квиз
Завршетак квиза	Уколико је корисник дао одговоре на сва питања, могуће је да заврши квиз. Након завршетка квиза, корисник има могућност провере питања на које је дао одговоре. Корисник добија титулу сходно броју тачних одговора у квизу.

5.3.4. Предности друштвене мреже Facebook

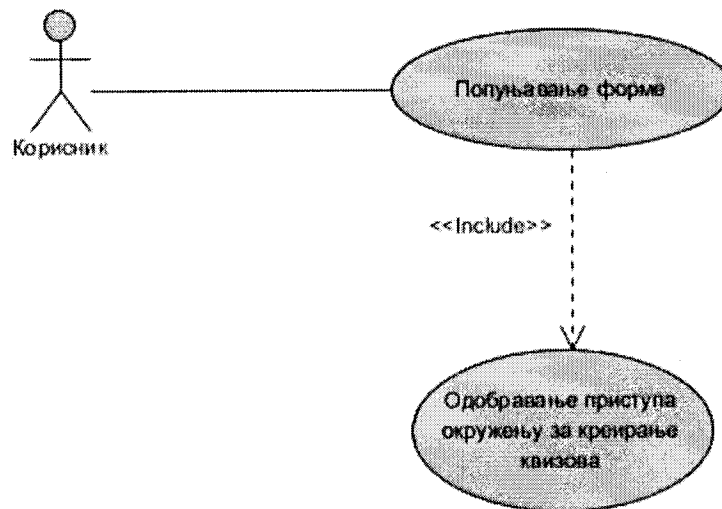


Слика 19: Дијаграм случаја употребе

Опис

Имена	Документација
Корисник	Корисник је свако ко има регистрован налог на друштвеној мрежи <i>Facebook</i>
Коришћење предности <i>Facebook</i> друштвене мреже	Предности <i>Facebook</i> друштвене мреже могу се на више начина искористити у самој апликацији. Након дозвола за читање података корисника могућности коју апликација има јесу вишеструке. У апликацији користиће се могућност дељења са пријатељима титуле након завршетка квиза и позивање пријатеља да тестирају своје знање из математике
Дељење са пријатељима титуле након завршетка квиза	Након завршетка квиза, корисник добија титулу за дати квиз и ту титулу може да подели са пријатељима уколико он то жели. Титула се додељује на основу тачних одговора, односно процентуалног броја које поена за дати квиз
Позивање пријатеља да тестирају своје знање из математике	Након завршетка квиза, корисник добија могућност да позове своје пријатеље на друштвеној мрежи <i>Facebook</i> да тестирају своје знање из математике, слањем захтева за прихватање апликације у домену <i>Facebook</i> друштвене мреже

5.3.5 Попуњавање форме за регистрацију професора

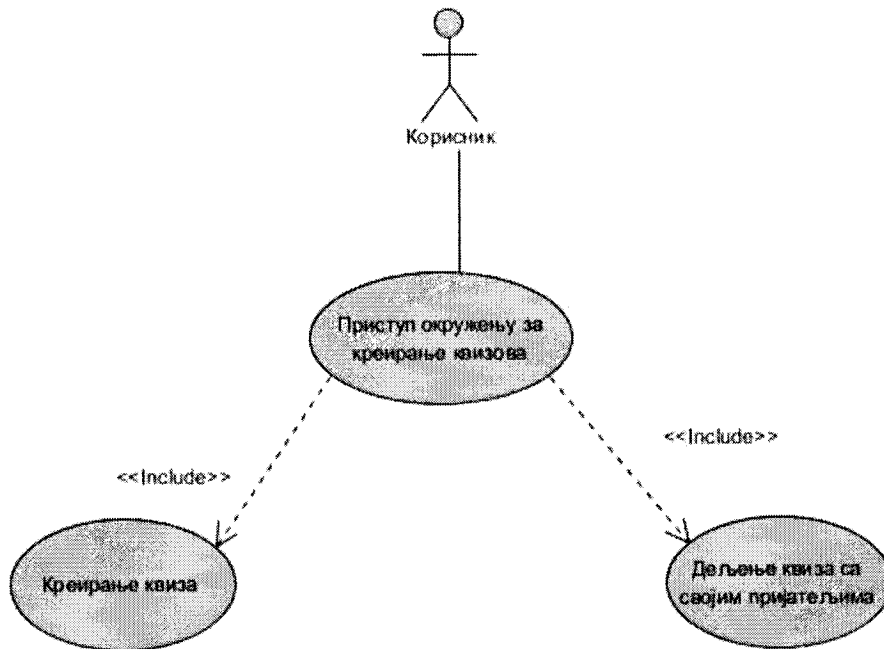


Слика 20: Дијаграм случаја употребе

Опис

Учесник	Документација
Корисник	Корисник је свако ко има регистрован налог на друштвеној мрежи <i>Facebook</i>
Попуњавање форме	Претпоставка је да корисник који има више од 22. године може да буде професор математике. Да би добио приступ окружењу за креирање квизова, потребно је да попуни форму за подацима. Подаци који се траже кориснику не морају да буду стандардни подаци на друштвеној мрежи <i>Facebook</i> . Због тога је направљена форма за „регистравање“ професора у самој апликацији.
Одобравање приступа окружењу за креирање квизова	Након слања података, потребно је кориснику одобрити приступ окружењу за креирање квизова. Одобравање приступа ради администратор апликације, јер је потребно проверити профил корисника који жели да добије приступ за креирање квизова. Због ограничења друштвене мреже <i>Facebook</i> , није могуће слање поруке кориснику.

5.3.6 Приступ корисничком окружењу

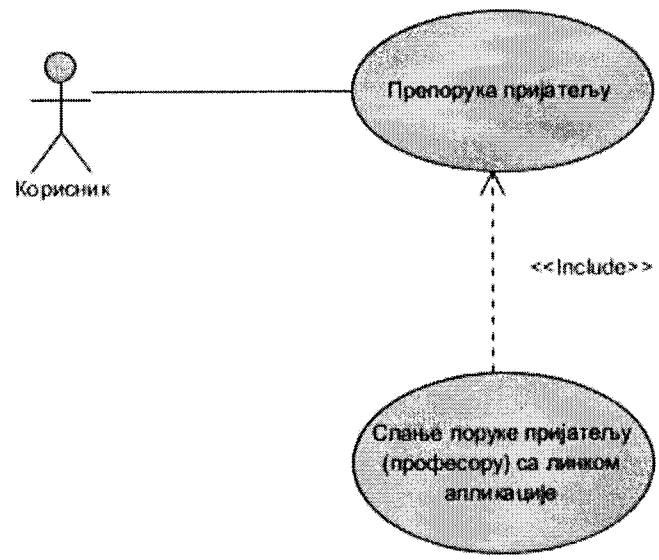


Слика 21. Дијаграм случаја употребе

Опис

Назив	Контентација
Корисник	Корисник је свако ко има регистрован налог на друштвеној мрежи <i>Facebook</i>
Приступ окружењу за креирање квизова	Ако је кориснику одобрен приступ окружењу за креирање квизова, онда се кориснику приликом приступања апликацији приказује дугме „Креирање квизова“. Кликом на дугме, кориснику се приказује окружење за креирање квизова
Креирање квиза	Сваки квиз треба да има назив, математичку област и разред. Професор добија форму коју треба да попуни са наведеним подацима. Сваки квиз треба да садржи 8 задатака различитих тежина и то требају да буду задаци за основни ниво знања (2), задаци за средњи ниво знања(4) и задаци за напредни нови знања (2).
Дељење квиза са својим пријатељима (ученицима)	Када се креира квиз, постоји могућност да се аутоматски подели са својим пријатељим (ученицима). Користи се додатак друштвене мреже <i>Facebook</i> .

5.3.7. Препорука пријатељу (професору)

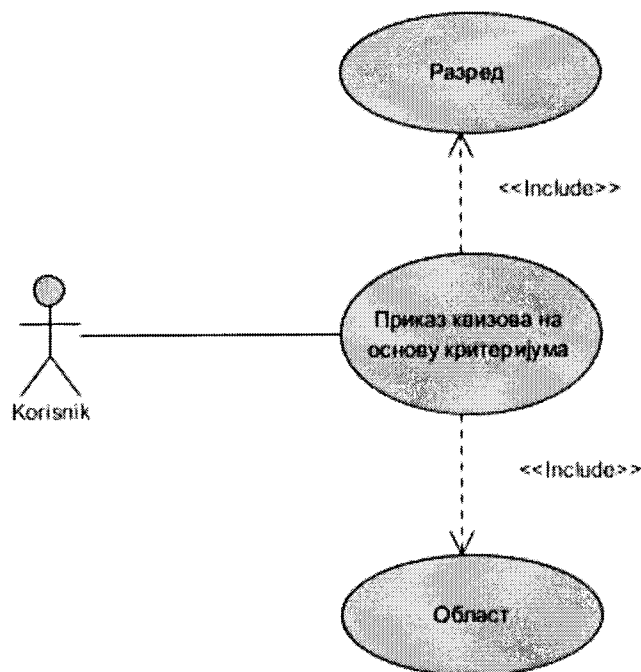


Слика 22. Дијаграм случаја употребе

Опис

Идентификатор	Опис
Корисник	Корисник је свако ко има регистрован налог на друштвеној мрежи <i>Facebook</i> .
Препорука пријатељу	Корисник који има мање од 22. године има могућност да препоручи апликацију свом професору. За препоруку користи се додатака друштвене мреже <i>Facebook</i> .
Слање поруке пријатељу (професору) са линком апликације	Ако корисник жели да препоручи апликацију неком свом пријатељу (свом професору математике), то може урадити кликом на дугме „Препоручите Вашем професору математике“. Назив дугмета није стриктно везан за препоруку професору математике, зато што није могуће препознати корисника да ли је то професор или не.

5.3.8. Одабир квизова на основу критеријума



Слика 23. Дијаграм случаја употребе

Опис

Документација	
Корисник	Корисник је свако ко има регистрован налог на друштвеној мрежи <i>Facebook</i>
Приказ квизова на основу критеријума	Корисник има могућност избора другог критеријума за приказ квизова. Критеријуми који су доступни кориснику су Разред и Област које треба да изабере из падајуће листе
Разред	Разред је први критеријум који је потребан за приказ квизова. Међутим, то није довољан критеријум, зато што је потребно да се изабере област за изабрани разред. Постоји вероватноћа да за одређени разред не постоје доступни квизови.
Област	Уколико за одређени разред постоје задаци у бази података, онда ће за дате задатке да се прикажу доступне области на основу који корисник може да изабере квизове који ће му бити приказани.

5.5. Имплементација апликације за тестирање математичког знања у домену друштвене мреже *Facebook*

У наставку рада биће објашњена израда апликације у домену друштвене мреже *Facebook*, технологије које су коришћене у изради апликације и развојно окружење. Да би неко могао да направи *Facebook* апликацију прво је потребно да има валидан профил на овој друштвеној мрежи. Потребно је искуство у изради веб страница (познавање *HTML*-а, *CSS*-а, *JavaScript*-а), познавање неког језика за израду динамичких веб апликација и познавање апликационих програмских интерфејса друштвене мреже *Facebook*. Потребно је да корисник региструје профил ка *developer*. Кораци за регистровање корисничког профила као *developer*-а и креирање апликације приказани су у додатку 2.

5.5.1. Технологије примењене у развоју апликације и развојно окружење

Програмске технологије коришћене у развоју апликације су карактеристичне и најраспрострањеније веб технологије данас. То су:

- *HTML*,
- *CSS*,
- *JavaScript*,
- *jQuery* [66],
- *jQueryUI* [67],
- *AJAX* [68],
- *PHP*,
- *SQL*

HTML (HyperText Markup Language)

HTML је стандардизовани језик који се користи при структурирању и опису елемената веб страница. *HTML* стандардизује и одржава *World Wide Web Consortium* (*W3C*). Једноставан је за употребу и брзо се учи што је један од разлога његове распрострањености и популарности. Сачињен је од обичног текста и знакова (енг. *tags*). Сврха *HTML* знакова је да се структура документа опише тако да би веб претраживач могао да је препозна и исправно прикаже.

На друштвеној мрежи *Facebook* раније је био доступан *FBML (Facebook Markup Language)* који се разликовао од стандардног *HTML*-а по томе што је поседовао специфичне знакове који су могли да се користе у апликацијама (пример: *fb:name*, *fb:comments* и други).

CSS (Cascading Style Sheets)

Језик који се користи при креирању веб страна је *CSS*. У основи, то је скуп инструкција које дефинишу како неки *HTML* документ треба да се прикаже. Може се дефинисати у спољашњој датотеци или фајлу. Веб претраживач може да прочита спољашњи фајл и примени га на одређену веб структуру стране. На тај начин се стил и форматирање стране одвајају од самог садржаја. Ово је врло битан концепт код дизајна. Пружа детаљан начин стилизовања веб апликација.

JavaScript

JavaScript припада групи језика за скриптовање, пре свега клијентске стране, мада се може извршавати и на серверу. Додаје интерактивност веб презентацији на

страни клијента. Коришћењем овог језика могуће је одговарати на акције корисника у самом веб претраживачу. Користи се са *HTML* језиком за креирање комплексни веб страна са богатим садржајем за корисника. Комбиновање *JavaScript*-а и *CSS* стилова, веб стране се могу представити на много различитих начина.

jQuery

jQuery је *JavaScript* библиотека која у себи има предефинисане функције које могу да се користе за управљањем *HTML* садржајем, догађајима и анимацијом и *AJAX* чини много једноставнијим и лакшим за употребу. То је библиотека отвореног кода која је веома распрострањена. Њена велика предност је подршка великог броја веб претраживача.

jQueryUI

jQueryUI је такође *JavaScript* библиотека која је зависна од *jQuery* библиотеке. И ова библиотека је отвореног кода која у себи има предефинисане графичке објекте који могу веома лако да се користе у изради веб апликација. И она је подржана од великог броја веб претраживача и веома лако може да промени стил неког од свог предефинисаног елемента.

AJAX (Asynchronous JavaScript and XML)

AJAX представља веб развојну технику за креирање интерактивних веб апликација. Ова технологија се заснива на размени малих количина података са сервером како корисник не би морао често да освежава страницу. На тај начин се омогућава потпуна интерактивност, ефикасност и побољшава функционисање веб апликације. *AJAX* није технологија сама по себи, већ термин који се односи на коришћење групе технологија. *AJAX* је базиран на интернет стандардима користећи комбинације:

- *XMLHttpRequest* (асинхроно прикупљање података)
- *JavaScript/DOM* (динамички приказ и интеракција преко *Document Object Model* и *JavaScript* који повезује све у целину)
- *HTML/CSS* (презентација базирана на стандардима *HTML* и *CSS*)
- *XML* (најчешћи формат за размену података)

AJAX апликације су независне у односу на претраживаче.

PHP (Hypertext Preprocessor)

PHP је скрипт језик који се користи за развој веб апликација на страни сервера. Он омогућава креирање динамичких и интерактивних веб апликација. *PHP* може да креира *HTML* страницу на серверу пре него што се она попуни динамичким садржајем и пошаље клијенту. *PHP* подржава објектну парадигму. Његова појава популаризовала је и стварање узорака (енг. *template*) који омогућавају такозвани брзи развој апликација (енг. *rapid application development*). То је језик отвореног кода. Подржава рад са различитим базама података. Компатибилан је са скоро свим серверима који се данас користе.

SQL (Structured Query Language)

SQL је стандардни језик за приступ и управљање базама података. Према *ANSI (American National Standards Institute)* то је стандардни језик за управљање релационим базама података. Главне *SQL* наредбе су: „*CREATE*“, „*ALTER*“, „*DROP*“. Најважније наредбе за обраду података су: „*SELECT*“, „*INSERT*“, „*UPDATE*“, „*DELETE*“.

Поред поменутих технологија коришћене су и специфичне за библиотеке у изради апликације и то:

- MathJax [69],
- TinyMCE [70]

MathJax

MathJax је JavaScript библиотека отвореног кода која пружа подршку за приказ математичких формула. То је пројекат *MathJax Consortium*, *American Mathematical Society* и *Society for Industrial and Applied Mathematics* који има циљ напредовање математике и других научних садржаја на вебу. Подржава *LaTeX*, *MathML* и друге језике за означавање формула у *HTML*-у. Поседује свој АПИ тако да се може лако интегрисати у веб апликацијама.

TinyMCE (Tiny Moxiecode Content Editor)

TinyMCE је JavaScript библиотека отвореног кода која пружа подршку за форматирање и уређивању текста при уносу података у базу података. За ову библиотеку постоји велики број додатака који могу да се интегришу. Један од тих додатака јесте додаток *Equation Editor* [71] који се користи за форматирање *Latex* математичких формула у задацима.

Развојно окружење

Развојно окружење које је коришћено у имплементацији апликације је:

- *NetBeans IDE* и
- *XAMPP*

NetBeans IDE (енг. *Integrated development environment*)

NetBeans је развојно окружење за *Java*, *PHP*, *JavaScript*, *HTML5*, *C*, *C++* и многе друге програмске језике. *NetBeans* написан је у *Java* програмском језику и може се покретати на свим оперативним системима који подржавају *JVM (Java Virtual Machine)*. То је окружење отвореног кода, које данас развија *Oracle*.

XAMPP

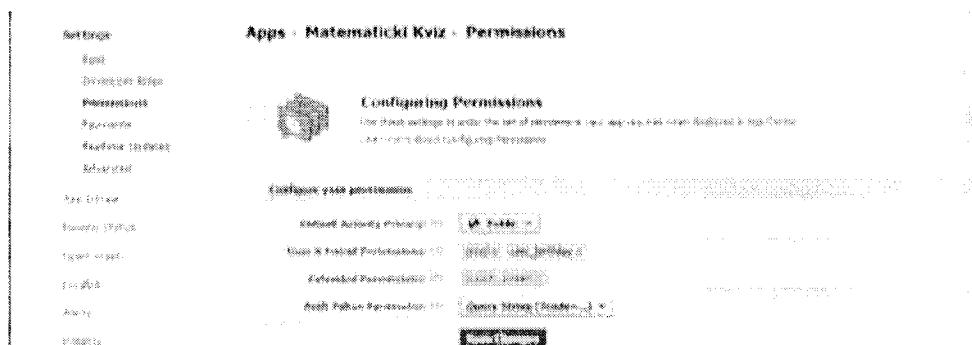
XAMPP је скуп програма за развој веб апликација. *XAMPP* чине *Apache* веб сервер, *MySQL* систем за управљањем релационим базама података, *PHP* програмски језик и *Perl* програмски језик.

5.5.2 Опис имплементације функционалних захтева

Друштвена мрежа *Facebook* омогућава апликацијама да приступају корисничким подацима уколико корисник користи неку од апликација на овој друштвеној мрежи. Приступ корисничким подацима је веома ограничен. Код подешавања апликације, потребно је навести којим корисничким подацима апликација приступа.

Апликација за тестирање математичког знања у домену друштвене мреже *Facebook* захтева од корисника дозволу за приступа подацима и то: *email*, *user_birthday*, *publish_stream*. Осталим корисничким подацима апликација нема права приступа. Наведени кориснички подаци користе се у апликацији за креирање јединственог

корисничког искуства коришћењем апликације за тестирање математичког знања. Овим корисничким подацима могуће је приступити уколико корисник „регистраје“ апликацију.

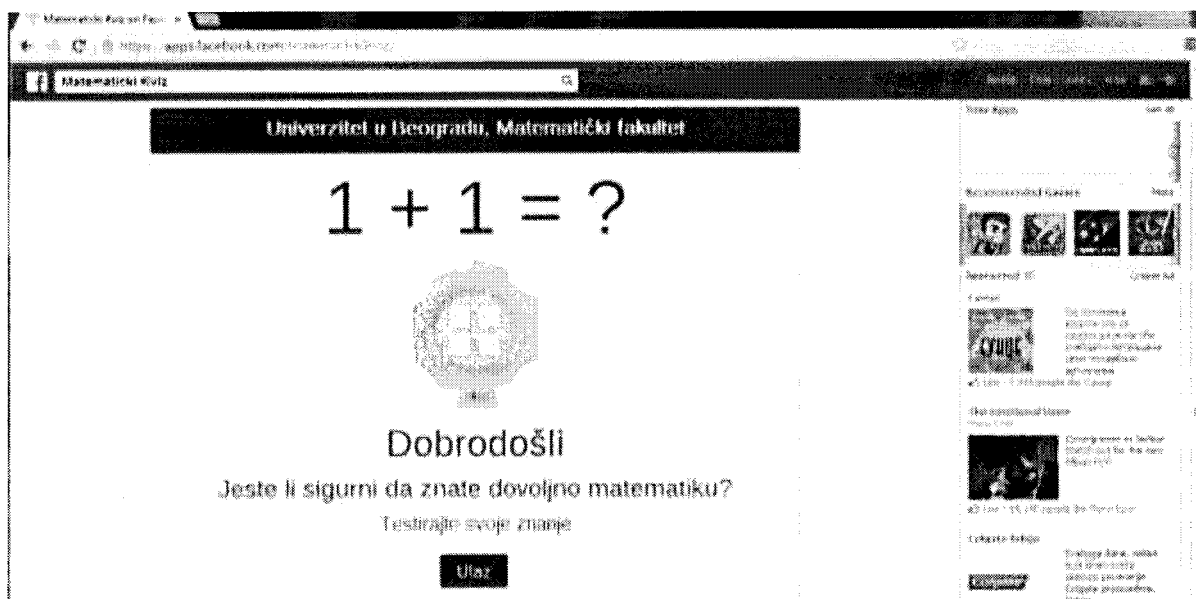


Слика 25. Подешавање дозвола (енг. *Permissions*) за приступ корисничким подацима

За ауторизацију апликације на корисничком профилу потребно је дефинисати дозволе при приступању апликацији. На почетној страни апликације корисник уколико жели да ауторизује апликацију на свом профилу, потребно је да прихвати захтеве које добија кликом да дугме „Улаз“. У наставку приказан је код за приказ почетне стране апликације, где је уграђена функција за ауторизацију апликације `$facebook->getLoginUrl()`.

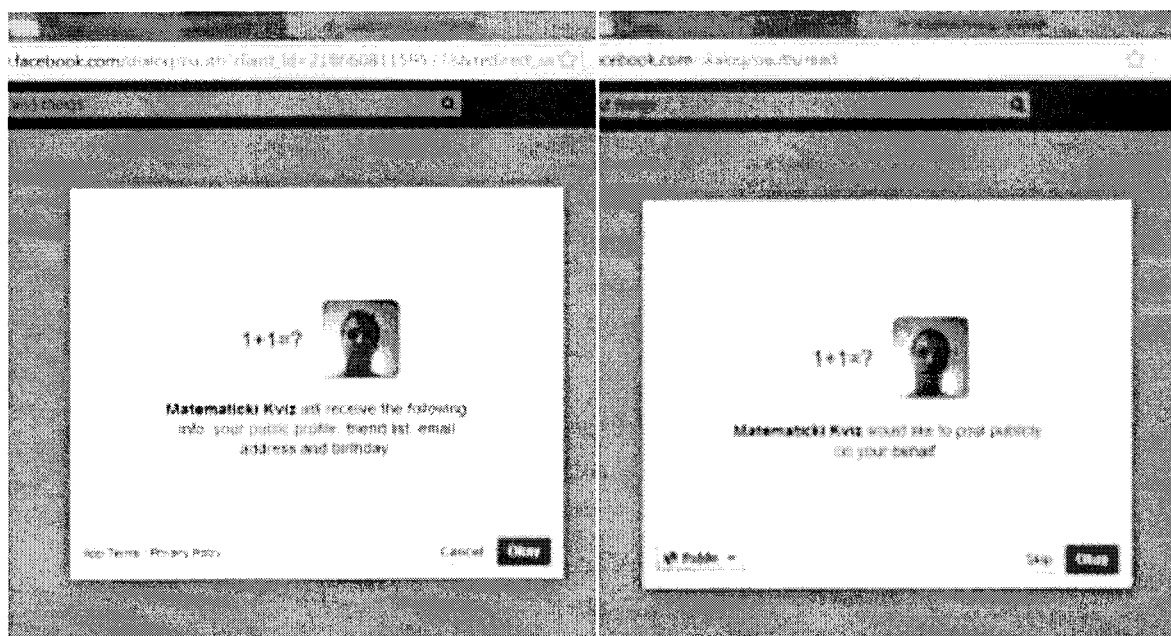
```
<div id="wrapper" class="ui-widget ui-widget-content">
  <div id="naziv_app">1 + 1 = ?</div>
  
  <div id="dobrodosli">Dobrodošli</div>
  <div id="parola1">Jeste li sigurni da znate dovoljno matematiku?</div>
  <div id="parola2">Testirajte svoje znanje</div>
  <div id="ulaz">
    <?php
      $redirect_uri = 'https://apps.facebook.com/matematickikviz/';
      $loginUrl = $facebook->getLoginUrl(array(
        'display'=>'popup',
        'scope'=>'email,user_birthday,publish_stream',
        'redirect_uri'=>$redirect_uri
      ));
      echo '<a href="' . $loginUrl . '" target="_top">Улаз</a>';
    ?>
  </div>
  <div id="copyright">Апликација је део master рада Ivana Milosavljevića</div>
</div>
```

Овде је приказан и део кода у ком се позива *Facebook PHP* АПИ. Наводе се дозволе за апликацију које се подешавају приликом креирања апликације.



Слика 26. Почетна страна апликације

Кликне на дугме „Улаз“ кориснику се приказује дијалог са захтевом приступа његовим подацима на овој друштвеној мрежи. (слика 32)



Слика 27. Захтев за приступ корисничким подацима и објављивање информација на корисниковом профилу

Уколико је ауторизована апликација, онда се кориснику приказују препоручени квизови који су генерисани на основу година корисника (слика 34).



Слика 28. Приказ апликације са профила корисника различитих година

Алгоритам за генерисање квизова на основу година корисника генерише квизове и приказује квизове кориснику. Године корисника се одређују на основу податка који се добија на овој друштвеној мрежи, а то је рођендански датум корисника. Алгоритам је имплементиран тако да корисник у највећи број случајева добије квизове које није решавао. То се проверава у табели `generisani_kviz` у којој се чувају сви квизови које је корисник решио, односно квизови за које постоји резултат. Генерисани квизови доступни су кориснику 24 сата након приступа апликацији. Ово ограничење направљено је због тога да би корисник након 24 сата увек добијао квизове за новим задацима. Увек за сваког корисника генерише се шест предлога квизова. Предлози квизова за кориснике се чувају у табели `predlozi_kvizova`. Приликом уноса предлога квизова за сваки квиз се генерише `hash` вредност квиза алгоритмом `md5` (енг. Message-Digest algorithm 5) [72] која јединствено одређује сваки квиз. Ниска карактера која служи као генератор `hash` вредности је јединствени идентификатор корисника на *Facebook*-у, јединствени идентификатор питања и категорије за дато питање. У наставку приказан је *PHP* код функције `generateStringForHash`.

```
public function generateStringForHash($FBUID,$quiz){
    $str =$FBUID;
    foreach ($quiz as $value) {
        foreach ($value as $questionCategory) {
            $str .= $questionCategory[0].$questionCategory[1];
        }
    }
    return $str;
}
```

Функција која генерише хеш вредност је:

```
public function generisiHashPredlogaKviza($stringFBUID_questions){
    return md5($stringFBUID_questions);
}
```

Ако се сви подаци генерисани алгоритмом налазе у базу података, онда се квизови приказују кориснику. Функција `checkQuizToDB` проверава да ли квизови за датог корисника на основу његових година се налазе у бази података. У наставку приказан је *PHP* код функције `checkQuizToDB`.

```
public function checkQuizToDB($FBUID, $birthday){
    global $db;
    global $qm;
    global $u;
    $db = Database::getInstance();
    $qm = QueryManager::getInstance();
    $u = Utillity::getInstance();
    $predloziKvizova = null;

    if(!$db->isConnected()){
        $db->connect();
    }

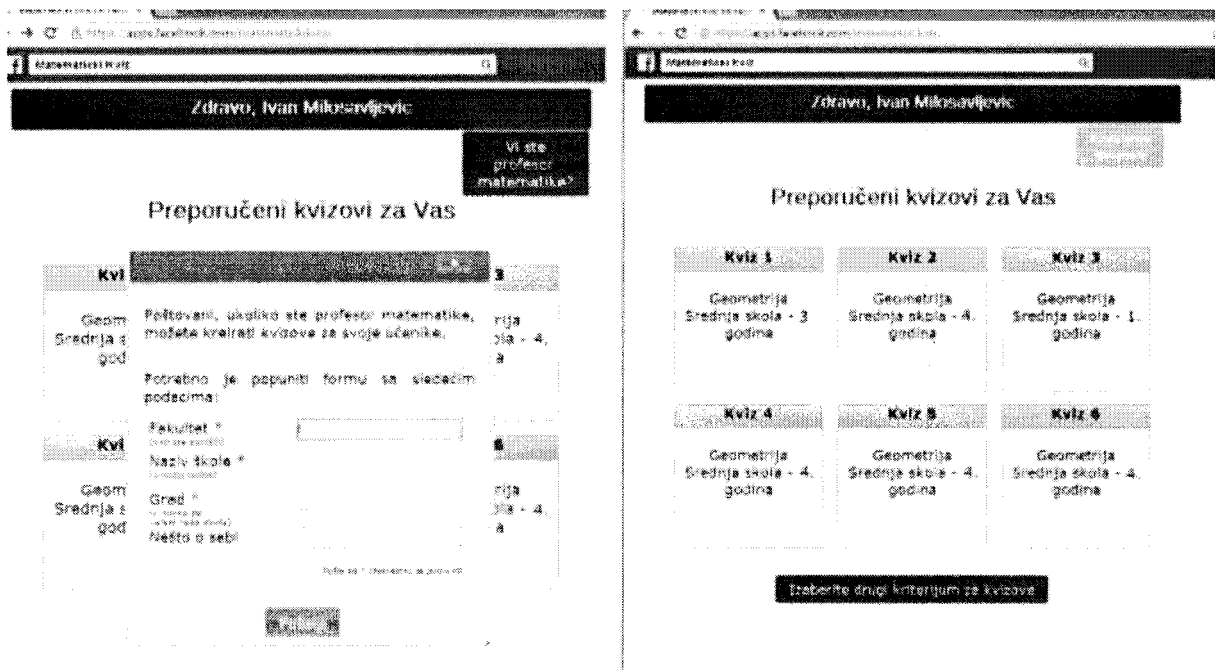
    $now = date('Y-m-d H:i:s');
    $minDatetimeDb = $qm->dbGetMinDatetimePK();

    if($qm->dbCheckExistsKvizovi($FBUID) &&
        !$qm->dbCheckExistsGenerisaniKvizove($FBUID)){

        if(! is_null($minDatetimeDb) &&
            strtotime($now) - strtotime($minDatetimeDb) > 86400 ){
            if($qm->dbDeletePredloziKvizovi()){
                return $this->addQuizToUser($FBUID,$birthday);
            }
        }
        else{
            if($qm->dbGetPredloziKvizova($FBUID, $predloziKvizova)){
                return $predloziKvizova;
            }
        }
    }
    else{

        if($qm->dbCheckExistsGenerisaniKvizove($FBUID) &&
            $qm->dbDeletePredloziKvizoviExistQuiz($FBUID)){
            return $this->addQuizToUser($FBUID,$birthday);
        }else{
            return $this->addQuizToUser($FBUID,$birthday);
        }
    }
}
```

Корисник који има више од 22 године, може да затражи приступ корисничком окружењу за креирање квизова. Кликком на дугме „Vi ste profesor matematike?“ отвара се дијалог у коме се налази форма која треба да се попуни како би корисник послао захтев за приступ корисничком окружењу.

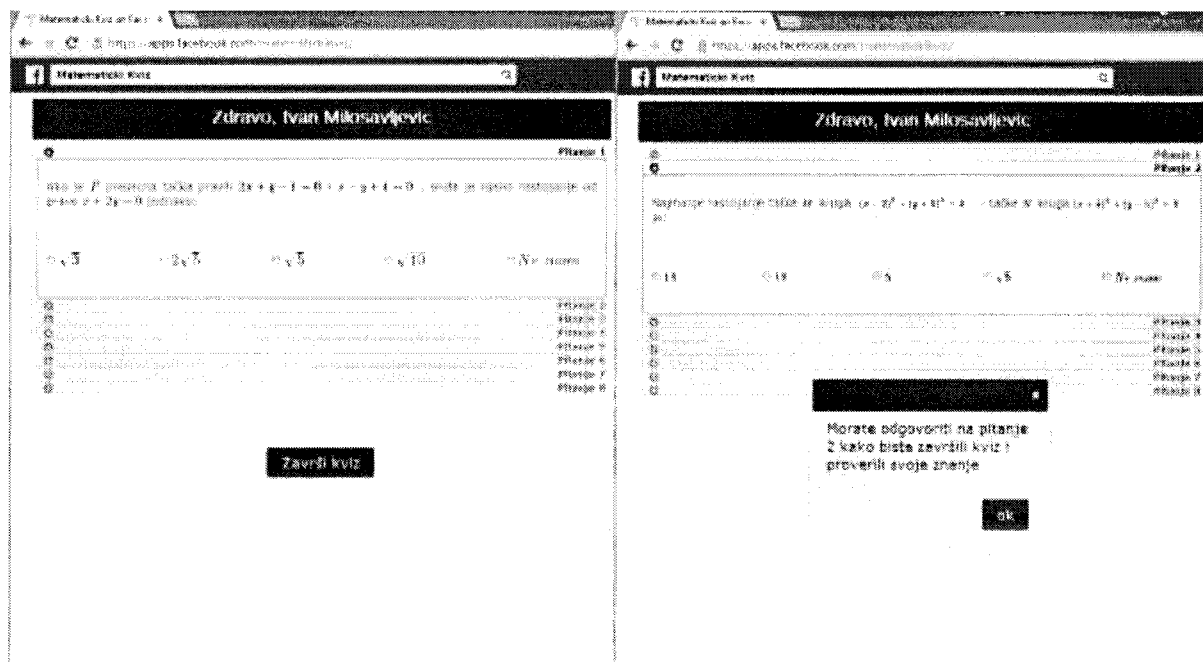


Слика 29. Форма за приступ корисничком окружењу за креирање квизова и приказ (лево) и приказ неодобреног захтева за приступ окружењу (десно)

Када корисник пошаље захтев, потребно је да се тај захтев и одобри. Администратор апликације одобрава захтев. Проверава се кориснички профил и подаци које је корисник унео у форму и након тога администратор одлучује да ли се кориснику одобрава приступ окружењу за креирање квизова или не. Подаци који се траже кориснику при попуњавању форме не постоје у оквиру сваког корисничког профила корисника на друштвеној мрежи *Facebook*.

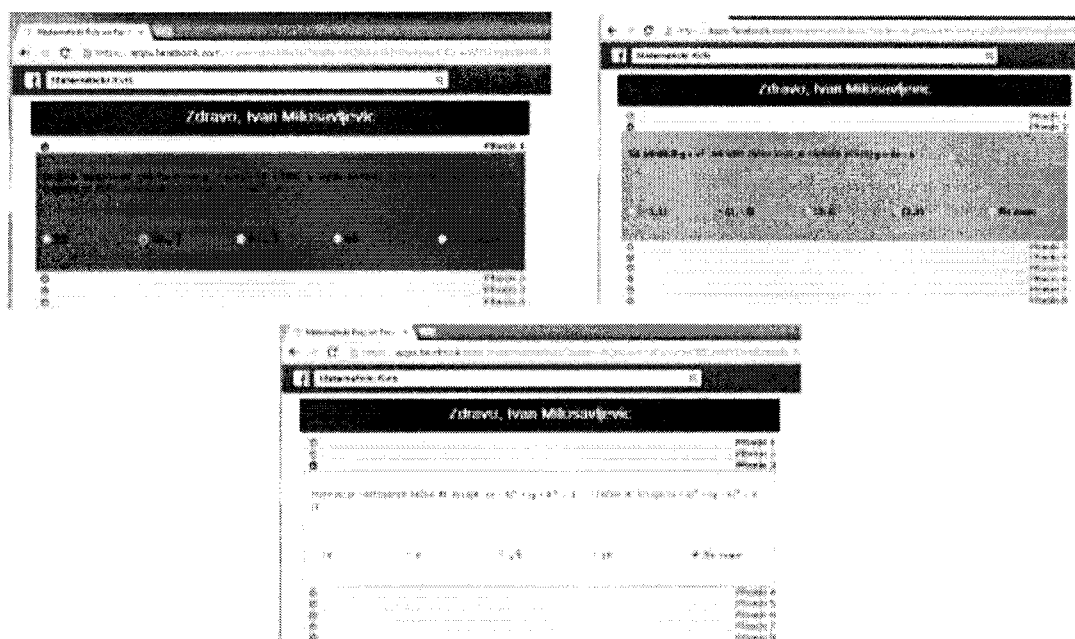
5.5.3 Опис имплементације решавања изабраног квиза

Корисник изабира неки од понуђених од квизова. Приказују се питања (задачи) за одабран квиз. Питање се састоји од текста и понуђених одговора. Један од четири понуђена одговора је тачан. Постоји и одговор „Не знам” за свако питање. Корисник треба да одговори на сва питања уколико жели да заврши квиз.



Слика 30. Решавање квиза

Кликом да дугме „Završi kviz“ врши се провера питања, да ли су дати сви одговори и уколико су сви одговори дати корисник добија титулу на основу датих одговора. Назив титуле зависи од броја тачних одговора. Свако питање се у зависности од тачности одговора боји се у различитим бојама и то, за тачан одговор то је зелена боја а за нетачан то је црвена боја. Уколико корисник није знао одговор на питање и „заокружио“ одговор „Ne znam“, питање не мења боју, односно то питање остаје беле боје. Свако питање вреди 12.5 поена.



Слика 31. Приказ питања након завршетка квиза

У *JavaScript* написана је функција за проверу одговора на питања и уколико су сви одговори дати онда се они шаљу серверу *AJAX* методом и одговор који се добија ажурирају се питања и приказују се резултати кориснику. Функција која се позива за проверу и слање одговора је `sendAnswers` и има аргумент `hash_kviza` који јединствено одређује квиз који корисник решава. У наставку приказан је *JavaScript* код функције `sendAnswers`.

```
function sendAnswers(hash_kviza){
    jQuery.noConflict();
    jQuery(function($) {
        var pitanje1 = $('input[name=pitanje1]:checked').val();
        var pitanje2 = $('input[name=pitanje2]:checked').val();
        var pitanje3 = $('input[name=pitanje3]:checked').val();
        var pitanje4 = $('input[name=pitanje4]:checked').val();
        var pitanje5 = $('input[name=pitanje5]:checked').val();
        var pitanje6 = $('input[name=pitanje6]:checked').val();
        var pitanje7 = $('input[name=pitanje7]:checked').val();
        var pitanje8 = $('input[name=pitanje8]:checked').val();

        if( pitanje1 == undefined){
            showDialogError('Morate odgovoriti na pitanje 1 kako biste završili kviz i proverili svoje znanje');
        }
        else if( pitanje2 == undefined){
            showDialogError('Morate odgovoriti na pitanje 2 kako biste završili kviz i proverili svoje znanje');
        }
        else if( pitanje3 == undefined){
            showDialogError('Morate odgovoriti na pitanje 3 kako biste završili kviz i proverili svoje znanje');
        }
        else if( pitanje4 == undefined){
            showDialogError('Morate odgovoriti na pitanje 4 kako biste završili kviz i proverili svoje znanje');
        }
        else if( pitanje5 == undefined){
            showDialogError('Morate odgovoriti na pitanje 5 kako biste završili kviz i proverili svoje znanje');
        }
        else if( pitanje6 == undefined){
            showDialogError('Morate odgovoriti na pitanje 6 kako biste završili kviz i proverili svoje znanje');
        }
        else if( pitanje7 == undefined){
            showDialogError('Morate odgovoriti na pitanje 7 kako biste završili kviz i proverili svoje znanje');
        }
        else if( pitanje8 == undefined){
            showDialogError('Morate odgovoriti na pitanje 8 kako biste završili kviz i proverili svoje znanje');
        }
    });
}
```

```

else{
    var data = {
        action: 'odgovori_na_pitanja_za_kviz',
        hash_kviza: hash_kviza,
        pitanje1: pitanje1,
        pitanje2: pitanje2,
        pitanje3: pitanje3,
        pitanje4: pitanje4,
        pitanje5: pitanje5,
        pitanje6: pitanje6,
        pitanje7: pitanje7,
        pitanje8: pitanje8
    };

    $.ajax({
        type : 'POST',
        url : ajax_url,
        data : data,
        beforeSend: function(){
            $("#infoSendLogin").html(htmlInfoSend);
            $("#infoSendLogin").dialog('open');
        },
        success : function (data){
            $("#infoSendLogin").dialog('close');
            if(!$.isEmptyObject(data)){

                var response = jQuery.parseJSON(data);
                addClassPitanje($(".pitanje1"),response.pitanje1);
                addClassPitanje($(".pitanje2"),response.pitanje2);
                addClassPitanje($(".pitanje3"),response.pitanje3);
                addClassPitanje($(".pitanje4"),response.pitanje4);
                addClassPitanje($(".pitanje5"),response.pitanje5);
                addClassPitanje($(".pitanje6"),response.pitanje6);
                addClassPitanje($(".pitanje7"),response.pitanje7);
                addClassPitanje($(".pitanje8"),response.pitanje8);

                removeZavrskiKvizBtn();

                showTitulaKorisnik(response.oblast_kategorija,response.titula,response.rezultat);
            }
        },
        error : function (data){
            console.log(data);
        }
    });
}
});
}

```

5.5.4 Опис имплементације коришћења АПИ библиотека *Facebook*

Коришћење АПИ библиотека друштвене мреже *Facebook* могу се искористити функције које библиотеке пружају. Да би се користиле функције АПИ библиотека, потребно је прво извршити иницијализацију библиотека. *PHP* АПИ библиотека иницијализује се са `AppID` и `AppSecret`. За коришћење *PHP* библиотеке потребно је проверити да ли се постоје инсталирани додаци за овај програмски језик и то: `cURL` и `JSON` јер без ових додатака није могуће користити ову библиотеку. У наставку приказан је код за иницијализацију *PHP* библиотеке.

```
require_once 'lib/fb-php-sdk/facebook.php';
global $facebook;
if(is_null($facebook)){
$facebook = new Facebook(array(
    'appId'=>218660811595378,
    'secret'=>'APP_SECRET_KEY',
    'cookie'=>true
));
```

За иницијализацију *JavaScript* АПИ библиотеке није потребан сигурносни кључ, већ је довољан само `AppID`. Унутар `body` тагу неопходно је убацити и `<div id="fb-root"></div>` за иницијализацију библиотеке. У наставку приказан је код за иницијализацију *JavaScript* библиотеке.

```
<script type="text/javascript">
    window.fbAsyncInit = function() {
        FB.init({
            appId : '218660811595378',
            frictionlessRequests: true,
            status: true,
            cookie : true
        });
    };
    (function(d, s, id){
        var js, fjs = d.getElementsByTagName(s)[0];
        if (d.getElementById(id)) {return;}
        js = d.createElement(s); js.id = id;
        js.src = "//connect.facebook.net/en_US/all.js";
        fjs.parentNode.insertBefore(js, fjs);
    })(document, 'script', 'facebook-jssdk');
```

```
</script>
```

Ако је иницијализација библиотека *PHP* и *JavaScript* успешна онда се могу користити функције које ове библиотеке поседују. Функције *PHP* библиотеке користе се при самом приступању апликацији, где се проверава да ли је тај корисник ауторизован корисник апликације. Позива се функција из *PHP* библиотеке `$facebook->getUser()`. Ако је корисник ауторизован корисник, позива се функција `$facebook->getAccessToken()` за добијање `access_token` који потребан за приступ корисничким

подацима преко *Graph* АПИ ове друштвене мреже. За позив ка *Graph* АПИ-у, потребан је валидан `access_token` за датог корисника. Ово је ниво сигурности ове друштвене мреже. Функција `isValidAccessToken` проверава да ли је `access_token` валидан и уколико јесте, добијају се подаци преко *Graph* АПИ-а. За нас најбитнији је податак о датуму рођења корисника како би се генерисао квиз на основу корисникових година. Уколико `access_token` није валидан, онда се кориснику приказује дијалог за ауторизацију апликације. У наставку приказан је *PHP* код функције `isValidAccessToken`.

```
public function isValidAccessToken($access_token, &$amp;decoded_response){
    $app_id = "218660811595378";
    $app_secret = "APP_SECRET_KEY";
    $my_url = "https://apps.facebook.com/matematickikviz/";

    $code = $_REQUEST["code"];

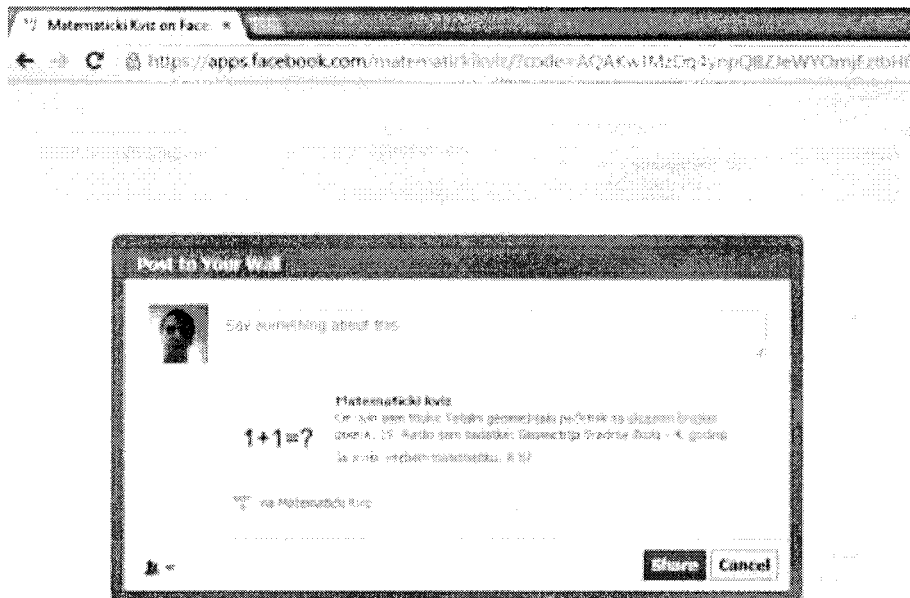
    if (isset($code)) {
        $token_url="https://graph.facebook.com/oauth/access_token?client_id="
            . $app_id . "&redirect_uri=" . urlencode($my_url)
            . "&client_secret=" . $app_secret
            . "&code=" . $code . "&display=popup";
        $response = file_get_contents($token_url);
        $params = null;
        parse_str($response, $params);
        $access_token = $params['access_token'];
    }
    $graph_url = "https://graph.facebook.com/me?"
        . "access_token=" . $access_token;
    $response = $this->curl_get_file_contents($graph_url);
    $decoded_response = json_decode($response);

    if ($decoded_response->error) {
        if ($decoded_response->error->type== "OAuthException") {

            $dialog_url= "https://www.facebook.com/dialog/oauth?"
                . "client_id=" . $app_id
                . "&redirect_uri=" . urlencode($my_url);
            echo("<script> top.location.href='" . $dialog_url
                . "'</script>");
            return false;
        }
        else {
            echo "other error has happened";
            return false;
        }
    }
    else {
        return true;
    }
}
```

Када корисник заврши решавање кви́за, онда се кориснику приказује титула на основу тачних одговора у кви́зу. Титулу за решаван квиз корисник може да објави на свом профилу. За објављивање титуле на корисничком профилу, користи се *JavaScript* АПИ библиотека и позива се функција `fb.ui` са методом `method: 'feed'`. Позивањем функције за објављивање титуле на корисничком профилу, отвара се дијалог са предефинисаном поруком која се односи на квиз који је корисник решавао. У наставку

приказан је дијалог за објаву титуле на корисничком профилу и код за приказ дијалога са предефинисаним текстом дијалога.

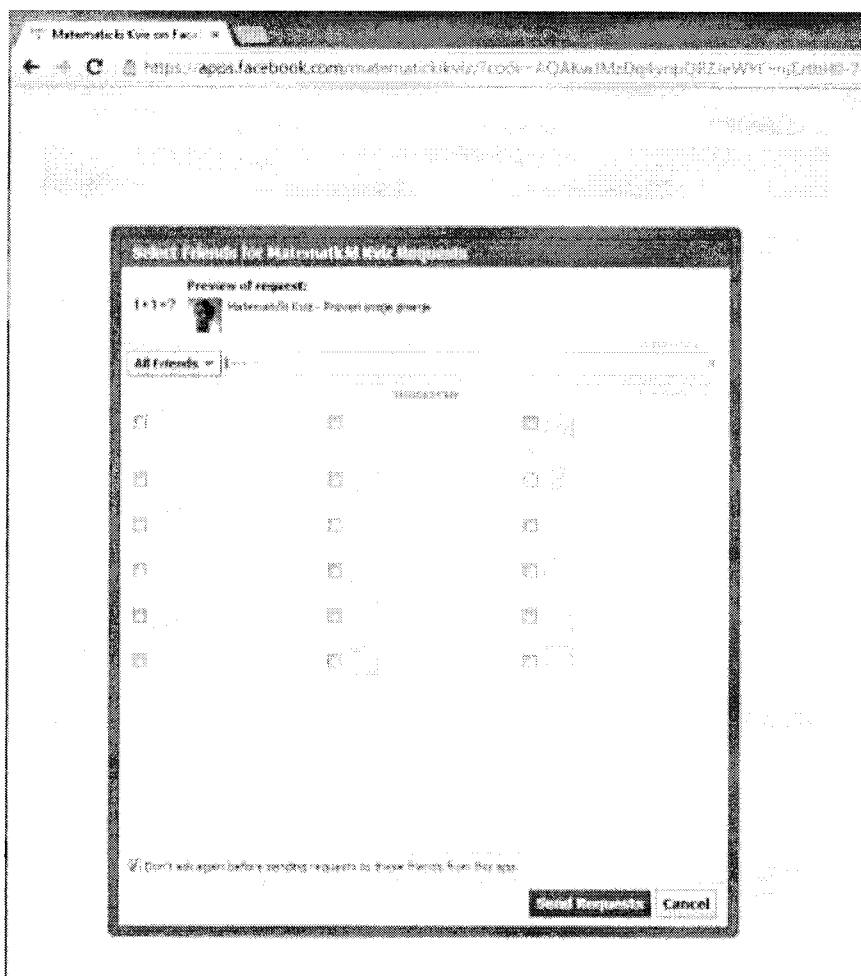


Слика 32. Објављивање титуле на корисничком профилу

```
function postToFeed(titula, brojPoena, oblast_kat) {
    var obj = {
        method: 'feed',
        redirect_uri: 'https://apps.facebook.com/matematickikviz/',
        link: 'https://apps.facebook.com/matematickikviz/',
        picture: 'https://muzikazaponeti.com/fb/quiz/images/ikona75x75.jpg',
        name: 'Matematički kviz',
        caption: 'Osvojio sam titulu: '+titula +' sa ukupnim brojem poena: '+brojPoena+'.
            Radio sam zadatke: '+oblast_kat,
        description: 'Ja ovde vezbam matematiku. A ti?'
    };

    function callback(response) {
        jQuery.noConflict();
        jQuery(function($) {
            if(response['post_id']){
                $('#titula_widget').fadeOut(1500);
            }
        });
    }
    FB.ui(obj, callback);
}
```

Корисник може својим пријатељима да препоручи апликацију. За препоруку апликације пријатељима такође се користи *JavaScript* АПИ библиотека и позива се функција `FB.ui` са методом `method: 'apprequests'`. Захтев се шаље пријатељима са предефинисаном поруком. У наставку приказан је дијалог за слање захтева пријатељима и код за приказ дијалога са предефинисаним текстом дијалога.



Слика 33. Слање захтева пријатељима за приступ апликацији

```
function sendRequestToFriends(){
    FB.ui({method: 'apprequests',
        message: 'Matemacki Kviz - Proveri svoje znanje'
    }, requestCallback);
}
```

5.6. Тестирање апликације у домену друштвене мреже *Facebook*

За тестирање апликације на друштвеној мрежи *Facebook* могуће је креирање тест корисника. Могуће је креирати кориснике који имају мање од 18 година. Креирање овакви корисника омогућило је проверу функција које генеришу квизове на основу корисникових година. Тестирање функционалности апликације рађено је током имплементације. Коришћени су алати за тестирање који се налазе на друштвеној мрежи *Facebook* у имплементацији. То су алати:

- <https://developers.facebook.com/tools/debug>
- <https://developers.facebook.com/tools/explorer/218660811595378/?method=GET>

6. Дискусија

Пријатељство у свакодневном животу се разликује од пријатељства на интернету [73]. У свакодневном животу пријатељ је најчешће особа са којом смо емотивно снажно повезани и за коју смо спремни да учинимо више него за остале. На интернету термин „пријатељ“ означава људе са којима сте повезани преко компјутера, било да су то пријатељи, познаници или непознати људи. На друштвеним мрежама „пријатељ“ је особа која има право приступа вашем профилу, тако да интернет пријатељи представљају избор личне публице.

Представљене друштвене мреже у раду имале су за циљ упознавање са појмом друштвених мрежа и могућност развијања апликација на некој о актуелних друштвених мрежа. Анализом тренутних актуелних друштвених мрежа како у свету, тако и у Србији, добио се закључак да је за израду апликације за тестирање математичког знања најпогоднија друштвена мрежа *Facebook*. Разлог одабирања друштвене мреже *Facebook* је тај што велики део популације у нашој земљи има активне профиле на овој друштвеној мрежи. Анализирајући апликационе програмске интерфејсе актуелних друштвених мрежа добијена је јасна слика о томе како и на који начин требају да се искористе погодности друштвених мрежа, односно друштвене мреже *Facebook*.

Методологије за развој софтвера приказане у раду представљају скуп методологија које се данас доста користе у софтверској индустрији. Не постоји савршена методологија за развој софтвера и у зависности о пројекта, примењује се једна или друга методологија развоја. Развој апликације (софтвера) за тестирање математичког знања у домену друштвене мреже *Facebook* није представљено стриктно једном методологијом развоја софтвера. Разлог због којег није примењена одређена методологија је природа пројекта и временски развој пројекта. Нису постојале стриктно дефинисане фазе због тога што се током развоја апликације мењала документације и начини имплементације. Међутим, идеја апликације за тестирање математичког знања у домену друштвених мрежа константно је унапређивана током развоја, тако да се може сматрати да је примењена агилна методологија за развој софтвера. Сам степен формализма који је примењен у писању документације приказан је на начин развоја софтвера код обједињеног развојног процеса.

Апликациони програмски интерфејси друштвене мреже *Facebook* за развој веб апликација омогућавају скуп функција које обезбеђују велики степен друштвених интеракција у апликацији, односно на овој друштвеној мрежи. Представљене функције *Facebook* АПИ-а у раду представљају део коришћених функција АПИ-а у самој апликацији које обезбеђују јединствено кориснички-друштвено искуство.

7. Закључак

Данас је човек изложен великој количини информација путем разних медија као што су телевизија, радио, новине и интернет. Друштвене мреже на интернету су незаобилазни део живота веома великог броја популације како у свету, тако и у нашој земљи. Циљ рада је био направити корисну и едукативну апликацију на некој од друштвених мрежа. Анализом друштвених мрежа добијен је закључак да је најпогоднија друштвена мрежа за развој апликације друштвена мрежа *Facebook*. Методологије за развој софтвера су незаобилазни део развоја неког софтвера, односно апликација. Анализа методологија омогућила је озбиљан приступ развоју апликације за тестирање математичког знања на платформи *Facebook*. Тренутна база података апликације садржи 175 питања и 700 одговора од који је 175 тачних које је унео аутор рада.

Апликација се може унапредити на више начина. Могуће је проширити функционалности апликације и то оптимизовати апликацију приказ за мобилним телефонима. Могуће је оптимизовати и направити апликације за *Android* платформу и за *iOS* платформу .

Литература

- [1] <http://reconstruction.eserver.org/064/boyd.shtml>
(приступао 23. септембар 2013. године)
- [2] Journal of Computer-Mediated Communication 13 (2008) 210-230 International Communication Association
- [3] <http://www.sixdegrees.com/> (приступао 12. јануара 2013. године)
- [4] <http://whatis.techtarget.com/definition/six-degrees-of-separation>
(приступао 23. септембар 2013. године)
- [5] <http://www.livejournal.com/> (приступао 12. јануара 2013. године)
- [6] <http://global.cyworld.com/> (приступао 23. септембар 2013. године)
- [7] <http://www.lunarstorm.se> (приступао 12. јануара 2013. године)
- [8] <http://www.ryze.com/> (приступао 15. јануара 2013. године)
- [9] <http://www.tribe.net> (приступао 15. јануара 2013. године)
- [10] <http://www.linkedin.com/> (приступао 15. јануара 2013. године)
- [11] <http://www.friendster.com/> (приступао 12. јануара 2013. године)
- [12] <http://www.match.com/> (приступао 12. јануара 2013. године)
- [13] <http://www.flickr.com/> (приступао 12. јануара 2013. године)
- [14] <http://www.last.fm/> (приступао 12. јануара 2013. године)
- [15] <http://www.youtube.com/> (приступао 12. јануара 2013. године)
- [16] <http://www.myspace.com> (приступао 21. јануара 2013. године)
- [17] http://www.princeton.edu/~achaney/tmve/wiki100k/docs/Metcalfe_s_law.html
(приступао 23. септембар 2013. године)
- [18] Online victimization of youth: Five years later. Report from Crimes Against Children Research Center, University of New Hampshire. Retrieved July 21, 2007 from <http://www.unh.edu/ccrc/pdf/CV138.pdf>
- [19] <http://www.twitter.com> (приступао 25. јануара 2013. године)
- [20] <https://twitter.com/about> (приступао 23. септембар 2013. године)
- [21] <http://plus.google.com> (приступао 26. јануара 2013. године)
- [22] <http://en.wikipedia.org/wiki/Google%2B> (приступао 26. јануара 2013. године)
- [23] <https://www.facebook.com> (приступао 07. фебруара 2013. године)
- [24] <http://www.socialbakers.com/facebook-statistics/>
(приступао 12. јануара 2013. године)
- [25] <http://www.socialbakers.com/facebook-statistics/serbia>
(приступао 18. фебруара 2013. године)
- [26] Statement on Professional Ethics, <http://www.aaup.org>
- [27] Faceworking: exploring students' education-related use of Facebook, Neil Selwyn, Learning, Media and Technology Vol. 34, No. 2, June 2009, 157–174
- [28] <http://www.techterms.com/definition/api> (приступао 23. септембар 2013. године)
- [29] AN ANALYSIS OF SOCIAL NETWORK CONNECT SERVICES, Antonio Tapiador, Víctor Sánchez and Joaquín Salvachúa, Universidad Politécnica de Madrid, Av. Complutense 30, Madrid, Spain {atapiador, vsanchez, jsalvachua} @dit.upm.es
- [30] <http://oauth.net/> (приступао 15. јун 2013. године)
- [31] <http://searchsoa.techtarget.com/definition/JavaScript>
(приступао 20. август 2013. године)
- [32] http://www.ics.uci.edu/~fielding/pubs/dissertation/rest_arch_style.htm
(приступао 23. септембар 2013. године)
















- [33] <http://www.json.org/> (приступао 23. септембар 2013. године)
- [34] Fielding, R., Gettys, J., Mogul, J., Frystyk, H., Masinter, L., Leach, P., and Berners-Lee, T. (1999). Hypertext Transfer Protocol – HTTP/1.1. RFC 2616 (Draft Standard). Updated by RFCs 2817, 5785, 6266.
- [35] <http://www.w3.org/TR/REC-xml/> (приступао 23. септембар 2013. године)
- [36] <http://www.w3schools.com/rss/> (приступао 23. септембар 2013. године)
- [37] <http://tools.ietf.org/html/rfc4287> (приступао 23. септембар 2013. године)
- [38] <https://developers.facebook.com/> (приступао 27. август 2013. године)
- [39] <https://developers.google.com/gdata/faq?hl=sr&csw=1#AuthScopes> (приступао 27. август 2013. године)
- [40] <http://www.apple.com/ios/what-is/> (приступао 27. август 2013. године)
- [41] <http://www.android.com/about/> (приступао 27. август 2013. године)
- [42] <http://php.net/> (приступао 27. август 2013. године)
- [43] <https://github.com/facebook/facebook-php-sdk> (приступао 28. август 2013. године)
- [44] A Comparative Analysis on Heavyweight V/s Lightweight Strategies of Software Development, Prof. Abhijit Khasnis Associate Professor, Tirupati Institute of Management
- [45] <http://agilemanifesto.org/> (приступао 23. септембар 2013. године)
- [46] <https://www.scrum.org/About> (приступао 23. септембар 2013. године)
- [47] <http://www-01.ibm.com/software/rational/rup/> (приступао 23. септембар 2013. године)
- [48] <http://best.berkeley.edu/~pps/pps/teams.html#what> (приступао 23. септембар 2013. године)
- [49] <http://www.scrum.org/Scrum-Guides> (Scrum Guide - RS.pdf) (приступао 01. март 2013. године)
- [50] A Comparison between Agile and Traditional Software Development Methodologies, M. A. Awad, The University of Western Australia, 2005
- [51] <http://www.ibm.com/developerworks/rational/library/769.html> (приступао 02. март 2013. године)
- [52] Release означава једну верзију испорученог софтверског производа.
- [53] Слика је преузета са сајта http://alberttwong.com/wp-content/uploads/2008/11/RUP/LargeProjects/#core.base_rup/guidances/supportingmaterials/welcome_2BC5187F.html (приступао 05. март 2013. године)
- [54] Glass' law: Glass, R.L.: Software Runaways. Lessons Learned from Massive Software Project Failures. Upper Saddle River, NJ: Prentice Hall 1998
- [55] Boehm's first law: Boehm, B.W., McClean, R.K., Urfrig, D.B.: Some Experience with Automated Aids to the Design of Large-Scale Reliable Software. IEEE Trans on Software Engineering 1, 1 (1975), 125–133
- [56] Boehm's second law: Boehm, B.W., Gray, T.E., Seewaldt, T.: Prototyping Versus Specifying: A Multiproject Experiment. IEEE Trans on Software Engineering 10, 3 (1984), 290–302
- [57] http://www.cs.scranton.edu/~jackowit/fall2012/se501/SelectedLaws_20061103.pdf (Beck-Fowler Hypothesis (84)) (приступао 06. март 2013. године)
- [58] McConnell, Steve. Software Estimation, Demystifying the Black Art. s.l. : Microsoft Press, 2006.
- [59] <http://bit.ly/LOx2c8> (приступао 09. март 2013. године)
- [60] International Journal of Computer Applications & Information Technology Vol. II, Issue I, January 2013 (ISSN: 2278-7720)



- [61] <http://www.visual-paradigm.com/> (приступао 15. април 2013. године)
- [62] <https://netbeans.org/> (приступао 15. април 2013. године)
- [63] <http://www.apachefriends.org/en/xampp-windows.html>
(приступао 15. април 2013. године)
- [64] <http://www.techterms.com/definition/sdk> (приступао 23. септембар 2013. године)
- [65] <http://www.digicert.com/ssl.htm> (приступао 23. септембар 2013. године)
- [66] <http://jquery.com/> (приступао 5. септембар 2013. године)
- [67] <http://jqueryui.com/> (приступао 5. септембар 2013. године)
- [68] http://www.w3schools.com/ajax/ajax_intro.asp
(приступао 23. септембар 2013. године)
- [68] <http://www.mathjax.org/> (приступао 5. септембар 2013. године)
- [70] <http://www.tinymce.com/index.php> (приступао 25. септембар 2013. године)
- [71] <http://www.codecogs.com/latex/about.php>
(приступао 25. септембар 2013. године)
- [72] <http://userpages.umbc.edu/~mabzug1/cs/md5/md5.html>
(приступао 25. септембар 2013. године)
- [73] <http://firstmonday.org/ojs/index.php/fm/article/view/1418/1336>
(приступао 25. септембар 2013. године)

Додатак 1.

Текстуални опис дијаграма базе података

Текстуални опис табела базе података

Назив	Опис
 korisnik	У овој табели налазе се основни подаци о кориснику који је дозволио приступ апликацији на Facebook друштвеној мрежи
 pitanje	У овој табели налазе се подаци која су питања за квиз који се предлаже кориснику.
 odgovori	У овој табели се налазе се подаци који су одговори у односу на дато питање
 kategorija_pitanja	У овој табели налазе се подаци о категорији питања
 oblast_pitanja	У овој табели налазе се подаци математичке области за које се генеришу квизови
 predlozi_pitanja_kvizova	У овој табели налазе се подаци о предлозима питања која се генеришу у односу на квиз који се приказује кориснику након дозволе корисника за приступ читања података апликацији
 predlozi_pitanja_profesor	У овој табели налазе се подаци о предлозима питања која се генеришу у односу на квиз који је креирао професор.
 predlozi_kvizova	У овој табели налазе се подаци о предлозима квизова који се генеришу кориснику након дозволе корисника за приступ читања података апликацији.
 predlozi_profesor	У овој табели се налазе се подаци о квизовима које је креирао професор у окружењу за креирање квизова
 generisani_kviz	У овој табели налазе се подаци о квизовима које је корисник завршио са одређеним резултатом
 pitanje_generisani_kviz	У овој табели налазе се подаци о питањима која се односе на квиз који је решавао корисник
 odgovori_korisnik	У овој табели налазе се подаци о одговорима на питања које је корисник „заокружио“ при завршетку квиза
 titula_korisnik	У овој табели налазе се подаци о корисницима који су решили неки од понуђених квизова и за њега добили одређену титулу
 titula	У овој табели налазе се подаци о титулама које су дефинисане у односу на број тачних одговора на квизу.
 uredjivac_pitanja	У овој табели налазе се подаци о професорима који су послали захтев за приступу окружењу за креирање квизова.


 log_activity_uredjivac_pitanja	У овој табели налазе се подаци о активностима које имају професори у окружењу за креирање квизова.
 privilegije	У овој табели налазе се подаци о привилегијама које имају професори који приступају окружењу за креирање квизова.


Текстуални опис табела базе података


Опис табеле korisnik

Приказ атрибута у физичком моделу података у табели korisnik

Назив	Тип података	Ограничења	Nullable
ID	bigint(20)	PK	No
FBUID	bigint(100)		No
ime	varchar(255)		No
prezime	varchar(255)		No
datumRodjenja	date		No
email	varchar(255)		No
app_access_token	varchar(1000)		No

fk_titulaKorisnik_korisnik : Релација међу табелама titula_korisnik и korisnik	
To	 titula_korisnik
On Delete	Cascade
On Update	Cascade
To Multiplicity	0..*
From Multiplicity	1
Data Model	Physical

fk_korisnik_predlozi : Релација међу табелама predlozi_kvizova и korisnik	
To	 predlozi_kvizova
On Delete	Cascade
On Update	Cascade
To Multiplicity	0..*
From Multiplicity	1
Data Model	Physical


fk_predlozi_profesor_korisnik : Релација међу табелама predlozi_profesor и korisnik	
To	 predlozi_profesor
On Delete	Cascade
On Update	Cascade
To Multiplicity	0..*
From Multiplicity	1
Data Model	Physical


 **Опис табеле pitanje**

Приказ атрибута у физичком моделу података код табеле pitanje


Назив	Тип података	Ограничења	Nullable
ID	bigint(20)	PK	No
kategoriyaPitanjaID	bigint(20)	FK (kategoriya_pitanja.ID)	No
oblastPitanjaID	bigint(20)	FK (oblast_pitanja.ID)	No
textPitanja	varchar(5000)		No


Релације објекта pitanje са другим табелама (физички приказ релација)


fk_odgovor_pitanje : Релација међу табелама odgovori и pitanje	
To	 odgovori
On Delete	Cascade
On Update	Cascade
To Multiplicity	0..*
From Multiplicity	0..1
Data Model	Physical


fk_pitanje_predlozi : Релација међу табелама predlozi_pitanja_kvizova и pitanje	
To	 predlozi_pitanja_kvizova
On Delete	Cascade
On Update	Cascade
To Multiplicity	0..*
From Multiplicity	1

Data Model	Physical
------------	----------

fk_kategorijaPitanja_pitanje : Релација међу табелама kategorija_pitanja и pitanje	
From	 kategorija_pitanja
On Delete	Cascade
On Update	Cascade
To Multiplicity	0..*
From Multiplicity	0..1
Data Model	Physical

fk_oblastPitanja_pitanje : Релација међу табелама oblast_pitanja и pitanje	
From	 oblast_pitanja
On Delete	Cascade
On Update	Cascade
To Multiplicity	0..*
From Multiplicity	1
Data Model	Physical

fk_predlozi_profesor_pitanja_pitanje : Релација међу табелама predlozi_profesor_pitanja и pitanje	
To	 predlozi_profesor_pitanja
On Delete	Cascade
On Update	Cascade
To Multiplicity	0..*
From Multiplicity	1
Data Model	Physical


 Опис табеле odgovori

Приказ атрибута у физичком моделу података код табеле odgovori

Назив	Тип података	Ограничења	Nullable
ID	bigint(20)	PK	No
pitanjeID	bigint(20)	FK (pitanje.ID)	No
odgovor	varchar(255)		No

<i>ispravan</i>	<i>int(1)</i>		<i>No</i>
<i>hash_odg</i>	<i>varchar(32)</i>		<i>No</i>

Релације објекта *odgovori* са другим табелама (физички приказ релација)


fk_odgovor_pitanje : Релација међу табелама <i>pitanje</i> и <i>odgovori</i>	
<i>From</i>	 <i>pitanje</i>
<i>On Delete</i>	<i>Cascade</i>
<i>On Update</i>	<i>Cascade</i>
<i>To Multiplicity</i>	<i>0..*</i>
<i>From Multiplicity</i>	<i>0..1</i>
<i>Data Model</i>	<i>Physical</i>


 Опис табеле *oblast_pitanja*

Приказ атрибута у физичком моделу података код табеле *oblast_pitanja*

Назив	Тип података	Ограничења	Nullable
<i>ID</i>	<i>bigint(20)</i>	<i>PK</i>	<i>No</i>
<i>naziv</i>	<i>varchar(255)</i>		<i>No</i>


Релације објекта *oblast_pitanja* са другим табелама (физички приказ релација)

fk_oblastPitanja_pitanje : Релација међу табелама <i>pitanje</i> и <i>oblast_pitanja</i>	
<i>To</i>	 <i>pitanje</i>
<i>On Delete</i>	<i>Cascade</i>
<i>On Update</i>	<i>Cascade</i>
<i>To Multiplicity</i>	<i>0..*</i>
<i>From Multiplicity</i>	<i>1</i>
<i>Data Model</i>	<i>Physical</i>

fk_titula_oblast_pitanja : Релација међу табелама <i>titula</i> и <i>oblast_pitanja</i>	
<i>To</i>	 <i>titula</i>
<i>On Delete</i>	<i>Cascade</i>
<i>On Update</i>	<i>Cascade</i>
<i>To Multiplicity</i>	<i>0..*</i>

<i>From Multiplicity</i>	1
<i>Data Model</i>	<i>Physical</i>

fk_predlozi_profesor_oblast_pitanja: Релација међу табелама predlozi_profeso и oblast_pitanja

<i>To</i>	 predlozi_profeso
<i>On Delete</i>	<i>Cascade</i>
<i>On Update</i>	<i>Cascade</i>
<i>To Multiplicity</i>	0..*
<i>From Multiplicity</i>	1
<i>Data Model</i>	<i>Physical</i>


 Опис табеле kategorija_pitanja

Приказ атрибута у физичком моделу података код табеле kategorija_pitanja

Назив	Тип података	Ограничења	Nullable
<i>ID</i>	<i>bigint(20)</i>	<i>PK</i>	<i>No</i>
<i>naziv</i>	<i>varchar(255)</i>		<i>No</i>
<i>godineOd</i>	<i>int(10)</i>		<i>No</i>
<i>godineDo</i>	<i>int(10)</i>		<i>No</i>
<i>tezina</i>	<i>int(10)</i>		<i>No</i>

Релације објекта kategorija_pitanja са другим табелама (физички приказ релација)

fk_kategorijaPitanja_pitanje : Релација међу табелама pitanje и kategorija_pitanja


<i>To</i>	 pitanje
<i>On Delete</i>	<i>Cascade</i>
<i>On Update</i>	<i>Cascade</i>
<i>To Multiplicity</i>	0..*
<i>From Multiplicity</i>	0..1
<i>Data Model</i>	<i>Physical</i>


 Опис табеле titula_korisnik


Приказ атрибута у физичком моделу података код табеле titula_korisnika


Назив	Тип података	Ограничења	Nullable
ID	bigint(20)	PK	No
TitulaID	bigint(20)	FK (titula.ID)	No
generisaniKvizID	bigint(20)	FK (generisani_kviz.ID)	No
korisnikID	bigint(20)	FK (korisnik.ID)	No

Релације објекта titula_korisnika са другим табелама (физички приказ релација)

fk_titula_titulaKorisnik – Релација међу табелама titula и titula_korisnik	
From	 titula
On Delete	Cascade
On Update	Cascade
To Multiplicity	0..*
From Multiplicity	1
Data Model	Physical

fk_titulaKorisnik_korisnik: Релација међу табелама korisnik и titula_korisnik	
From	 korisnik
On Delete	Cascade
On Update	Cascade
To Multiplicity	0..*
From Multiplicity	1
Data Model	Physical


fk_generisanKviz_titulaKorisnik : Релација међу табелама generisani_kviz и titula_korisnik	
From	 generisani_kviz
On Delete	Cascade
On Update	Cascade
To Multiplicity	0..*
From Multiplicity	1
Data Model	Physical


 Опис табеле titula

Приказ атрибута у физичком моделу података код табеле titula

Назив	Тип података	Ограничења	Nullable
ID	bigint(20)	PK	No
oblast_pitanjaID	bigint(20)	FK (oblast_pitanja.ID)	No
naziv	varchar(255)		No
procenat_tacnih_odgovora	double		No

Релације објекта titula са другим табелама (физички приказ релација)

fk_titula_titulaKorisnik : Релација међу табелама titula_korisnik и titula	
To	 titula_korisnik
On Delete	Cascade
On Update	Cascade
To Multiplicity	0..*
From Multiplicity	1
Data Model	Physical


fk_titula_oblast_pitanja : Релација међу табелама oblast_pitanja и titula	
From	 oblast_pitanja
On Delete	Cascade
On Update	Cascade
To Multiplicity	0..*
From Multiplicity	1
Data Model	Physical


 Опис табеле predlozi_pitanja_kvizova

Приказ атрибута у физичком моделу података код табеле predlozi_pitanja_kvizova

Назив	Тип података	Ограничења	Nullable
ID	bigint(255)	PK	No
pitanjeID	bigint(20)	FK (pitanje.ID)	No
predlozi_kvizova_ID	bigint(100)	FK (predlozi_kvizova.ID)	No

Релације објекта predlozi_pitanja_kvizova са другим табелама (физички приказ релација)

fk_pitanje_predlozi : Релација међу табелама pitanje и predlozi_pitanja_kvizova	
<i>From</i>	 pitanje
<i>On Delete</i>	<i>Cascade</i>
<i>On Update</i>	<i>Cascade</i>
<i>To Multiplicity</i>	0..*
<i>From Multiplicity</i>	1
<i>Data Model</i>	<i>Physical</i>


fk_predlozi_ppk : Релација међу табелама predlozi_kvizova и predlozi_pitanja_kvizova	
<i>From</i>	 predlozi_kvizova
<i>On Delete</i>	<i>Cascade</i>
<i>On Update</i>	<i>Cascade</i>
<i>To Multiplicity</i>	0..*
<i>From Multiplicity</i>	1
<i>Data Model</i>	<i>Physical</i>

 Опис табеле predlozi_pitanja_profesor

Приказ атрибута у физичком моделу података код табеле predlozi_pitanja_profesor


Назив	Тип података	Ограничења	Nullable
<i>ID</i>	<i>bigint(255)</i>	<i>PK</i>	<i>No</i>
<i>pitanjeID</i>	<i>bigint(20)</i>	<i>FK (pitanje.ID)</i>	<i>No</i>
<i>predlozi_profesorID</i>	<i>bigint(100)</i>	<i>FK (predlozi_profesor.ID)</i>	<i>No</i>

Релације објекта predlozi_pitanja_profesor са другим табелама (физички приказ релација)

fk_predlozi_profesor_ppp: Релација међу табелама predlozi_profesor и predlozi_pitanja_profesor	
<i>From</i>	 predlozi_profesor
<i>On Delete</i>	<i>Cascade</i>
<i>On Update</i>	<i>Cascade</i>

To Multiplicity	0..*
From Multiplicity	1
Data Model	Physical

fk_predlozi_profesor_pitanja_pitanje: Релација међу табелама pitanje и predlozi_pitanja_profesor

From	 pitanje
On Delete	Cascade
On Update	Cascade
To Multiplicity	0..*
From Multiplicity	1
Data Model	Physical


 Опис табеле predlozi_kvizova

Приказ атрибута у физичком моделу података код табеле predlozi_kvizova


Назив	Тип података	Ограничења	Nullable
ID	bigint(100)	PK	No
korisnikID	bigint(20)	FK (korisnik.ID)	No
hash_kviza	varchar(32)		No
datum_generisanja	datetime		No


Релације објекта predlozi_kvizova са другим табелама (физички приказ релација)

fk_predlozi_ppk : Релација међу табелама predlozi_pitanja_kvizova и predlozi_kvizova

To	 predlozi_pitanja_kvizova
On Delete	Cascade
On Update	Cascade
To Multiplicity	0..*
From Multiplicity	1
Data Model	Physical

fk_korisnik_predlozi : Релација међу табелама korisnik и predlozi_kvizova

From	 korisnik
On Delete	Cascade
On Update	Cascade
To Multiplicity	0..*
From Multiplicity	1
Data Model	Physical


 **Опис табеле predlozi_profesor**

Приказ атрибута у физичком моделу података код табеле predlozi_kvizova


Назив	Тип података	Ограничења	Nullable
ID	bigint(100)	PK	No
korisnikID	bigint(20)	FK (korisnik.ID)	No
oblast_pitanjaID	bigint(20)	FK (oblast_pitanja.ID)	No
naziv_kviza	varchar(32)		No
razred	varchar(255)		No
hash_kviza	varchar(32)		No
datum_generisanja	datetime		No

Релације објекта predlozi_kvizova са другим табелама (физички приказ релација)


fk_predlozi_profesor_ppp: Релација међу табелама predlozi_pitanja_profesor и predlozi_profesor

To	 predlozi_pitanja_profesor
On Delete	Cascade
On Update	Cascade
To Multiplicity	0..*
From Multiplicity	1
Data Model	Physical

fk_predlozi_profesor_k: Релација међу табелама korisnik и predlozi_profesor

From	 korisnik
On Delete	Cascade
On Update	Cascade

To Multiplicity	0..*
From Multiplicity	1
Data Model	Physical


fk_predlozi_profesor_op: Релација међу табелама oblast_pitanja и predlozi_profesor	
From	 oblast_pitanja
On Delete	Cascade
On Update	Cascade
To Multiplicity	0..*
From Multiplicity	1
Data Model	Physical


 Опис табеле generisani_kviz

Приказ атрибута у физичком моделу података код табеле generisani_kviz

Назив	Тип података	Ограничења	Nullable
ID	bigint(20)	PK	No
datumGenerisanjaKviza	datetime		No
hash_gen_kviz	varchar(32)		No
rezultat	double		Yes

Релације објекта generisani_kviz са другим табелама (физички приказ релација)

fk_generisaniKviz_pitanjaZaGenerisanKviz: Релација међу табелама pitanje_generisani_kviz и generisani_kviz	
To	 pitanje_generisani_kviz
On Delete	Cascade
On Update	Cascade
To Multiplicity	0..*
From Multiplicity	1
Data Model	Physical

fk_generisanKviz_titulaKorisnik : Релација међу табелама titula_korisnik и generisani_kviz	
To	 titula_korisnik


<i>On Delete</i>	<i>Cascade</i>
<i>On Update</i>	<i>Cascade</i>
<i>To Multiplicity</i>	<i>0..*</i>
<i>From Multiplicity</i>	<i>1</i>
<i>Data Model</i>	<i>Physical</i>


 **Опис табеле pitanje_generisani_kviz**

Приказ атрибута у физичком моделу података код табеле pitanje_generisani_kviz


Назив	Тип података	Ограничења	Nullable
<i>ID</i>	<i>bigint(20)</i>	<i>PK</i>	<i>No</i>
<i>pitanjeID</i>	<i>bigint(20)</i>	<i>FK (pitanje.ID)</i>	<i>No</i>
<i>generisaniKvizID</i>	<i>bigint(20)</i>	<i>FK (generisani_kviz.ID)</i>	<i>No</i>


Релације објекта pitanje_generisani_kviz са другим табелама (физички приказ релација)

fk_pitanja_odgovori_k : Релација међу табелама odgovori_korisnik и pitanje_generisani_kviz	
<i>To</i>	 <i>odgovori_korisnik</i>
<i>On Delete</i>	<i>Cascade</i>
<i>On Update</i>	<i>Cascade</i>
<i>To Multiplicity</i>	<i>0..*</i>
<i>From Multiplicity</i>	<i>1</i>
<i>Data Model</i>	<i>Physical</i>

fk_pitanje_pitanjeGenerisanoZaKviz : Релација међу табелама pitanje и pitanje_generisani_kviz	
<i>From</i>	 <i>pitanje</i>
<i>On Delete</i>	<i>Cascade</i>
<i>On Update</i>	<i>Cascade</i>
<i>To Multiplicity</i>	<i>0..*</i>
<i>From Multiplicity</i>	<i>0..1</i>
<i>Data Model</i>	<i>Physical</i>

fk_generisaniKviz_pitanjaZaGenerisanKviz: Релација међу табелама generisani_kviz и pitanje_generisani_kviz

<i>From</i>	 generisani_kviz
<i>On Delete</i>	<i>Cascade</i>
<i>On Update</i>	<i>Cascade</i>
<i>To Multiplicity</i>	<i>0..*</i>
<i>From Multiplicity</i>	<i>1</i>
<i>Data Model</i>	<i>Physical</i>


 **Опис табеле odgovori_korisnik**

Приказ атрибута у физичком моделу података код табеле odgovori_korisnik


Назив	Тип података	Ограничења	Nullable
<i>ID</i>	<i>bigint(20)</i>	<i>PK</i>	<i>No</i>
<i>korisnikID</i>	<i>bigint(20)</i>	<i>FK (korisnik.ID)</i>	<i>No</i>
<i>pitanje_generisani_kvizID</i>	<i>bigint(20)</i>	<i>FK (pitanje_generisani_kviz.ID)</i>	<i>No</i>
<i>ispravan</i>	<i>int(1)</i>		<i>No</i>

Релације објекта odgovori_korisnik са другим табелама (физички приказ релација)


fk_odgovoriKorisnik_korisnik : Релација међу табелама korisnik и odgovori_korisnik

<i>From</i>	 korisnik
<i>On Delete</i>	<i>Cascade</i>
<i>On Update</i>	<i>Cascade</i>
<i>To Multiplicity</i>	<i>0..*</i>
<i>From Multiplicity</i>	<i>1</i>
<i>Data Model</i>	<i>Physical</i>

fk_pitanja_odgovori_k : Релација међу табелама pitanje_generisani_kviz и odgovori_korisnik

<i>From</i>	 pitanje_generisani_kviz
<i>On Delete</i>	<i>Cascade</i>
<i>On Update</i>	<i>Cascade</i>
<i>To Multiplicity</i>	<i>0..*</i>
<i>From Multiplicity</i>	<i>1</i>


<i>Data Model</i>	<i>Physical</i>
-------------------	-----------------


 Опис табеле uredjivac_pitanja

Приказ атрибута у физичком моделу података код табеле uredjivac_pitanja


Назив	Тип података	Ограничења	Nullable
<i>ID</i>	<i>bigint(20)</i>	<i>PK</i>	<i>No</i>
<i>korisnikID</i>	<i>bigint(20)</i>	<i>FK (korisnik.ID)</i>	<i>No</i>
<i>privilegijeID</i>	<i>bigint(20)</i>	<i>FK (privilegije.ID)</i>	<i>No</i>
<i>odobren</i>	<i>int(1)</i>		<i>No</i>
<i>fakultet</i>	<i>varchar(255)</i>		<i>No</i>
<i>skola</i>	<i>varchar(255)</i>		<i>No</i>
<i>grad</i>	<i>varchar(255)</i>		<i>No</i>
<i>nesto_o_sebi</i>	<i>varchar(255)</i>		<i>Yes</i>

Релације објекта uredjivac_pitanja са другим табелама (физички приказ релација)

fk_log_uredjivac_pitanja: Релација међу табелама log_activity_uredjivac_pitanja и uredjivac_pitanja	
<i>To</i>	 log_activity_uredjivac_pitanja
<i>On Delete</i>	<i>Cascade</i>
<i>On Update</i>	<i>Cascade</i>
<i>To Multiplicity</i>	<i>0..*</i>
<i>From Multiplicity</i>	<i>1</i>
<i>Data Model</i>	<i>Physical</i>

fk_uredjivacPitanja_korisnik: Релација међу табелама korisnik и uredjivac_pitanja	
<i>From</i>	 korisnik
<i>On Delete</i>	<i>Cascade</i>
<i>On Update</i>	<i>Cascade</i>
<i>To Multiplicity</i>	<i>0..*</i>
<i>From Multiplicity</i>	<i>1</i>
<i>Data Model</i>	<i>Physical</i>

fk_uredjivacPitanja_privilegije: Релација међу табелама privilegije и uredjivac_pitanja

<i>From</i>	 privilegije
<i>On Delete</i>	<i>Cascade</i>
<i>On Update</i>	<i>Cascade</i>
<i>To Multiplicity</i>	<i>0..*</i>
<i>From Multiplicity</i>	<i>1</i>
<i>Data Model</i>	<i>Physical</i>


 **Опис табеле log_activity_uredjivac_pitanja**

Приказ атрибута у физичком моделу података код табеле uredjivac_pitanja

Назив	Тип података	Ограничења	Nullable
<i>ID</i>	<i>bigint(20)</i>	<i>PK</i>	<i>No</i>
<i>uredjivac_pitanjaID</i>	<i>bigint(20)</i>	<i>FK (uredjivac_pitanja.ID)</i>	<i>No</i>
<i>akcija</i>	<i>varchar(255)</i>		<i>No</i>
<i>akcija_datum</i>	<i>datetime</i>		<i>No</i>

Релације објекта uredjivac_pitanja са другим табелама (физички приказ релација)

fk_log_uredjivac_pitanja: Релација међу табелама uredjivac_pitanja и log_activity_uredjivac_pitanja


<i>From</i>	 uredjivac_pitanja
<i>On Delete</i>	<i>Cascade</i>
<i>On Update</i>	<i>Cascade</i>
<i>To Multiplicity</i>	<i>0..*</i>
<i>From Multiplicity</i>	<i>1</i>
<i>Data Model</i>	<i>Physical</i>

 **Опис табеле privilegije**

Приказ атрибута у физичком моделу података код табеле privilegije

Назив	Тип података	Ограничења	Nullable
<i>ID</i>	<i>bigint(20)</i>	<i>PK</i>	<i>No</i>
<i>Naziv</i>	<i>varchar(255)</i>		<i>No</i>

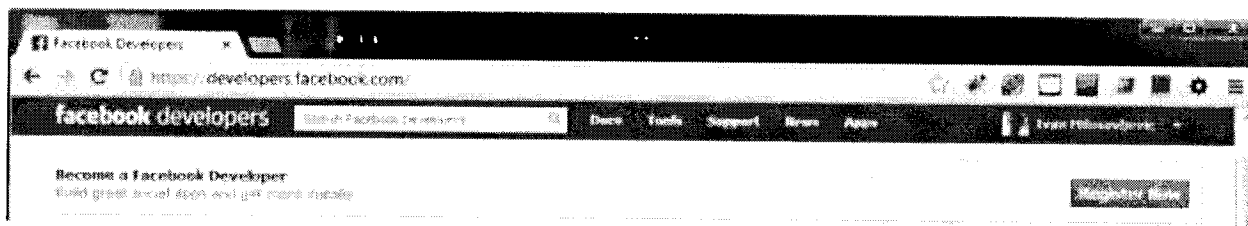
fk_uredjivacPitanja_privilegije: Релација међу табелама privilegije и uredjivac_pitanja

<i>To</i>	 uredjivac_pitanja
<i>On Delete</i>	<i>Cascade</i>
<i>On Update</i>	<i>Cascade</i>
<i>To Multiplicity</i>	<i>0..*</i>
<i>From Multiplicity</i>	<i>1</i>
<i>Data Model</i>	<i>Physical</i>

Додатак 2.

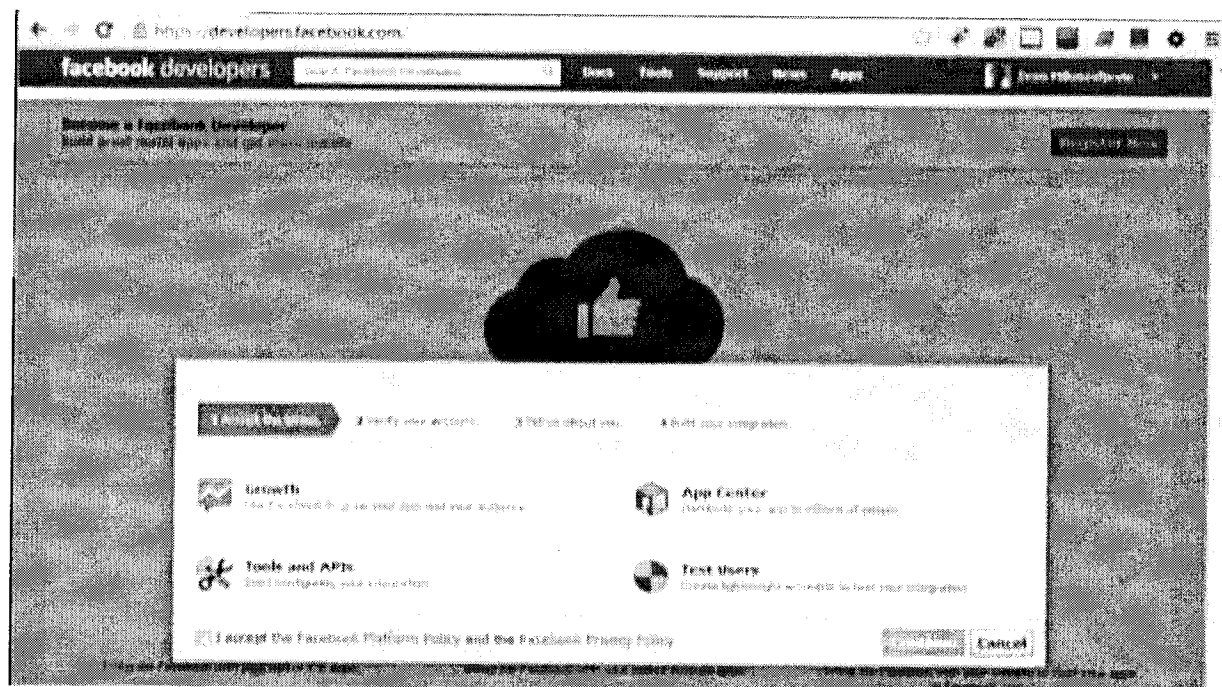
Регистровање Facebook developer

Facebook као платформа за развој апликације у свом домену нуди различите могућности креирања апликације. Одласком на адресу <https://developers.facebook.com/> потребно је регистровати се као *developer* на Facebook-у. Регистровање је просто, кликом на дугме *Register now* (слика 24).



Слика 34: Регистровање Facebook developer

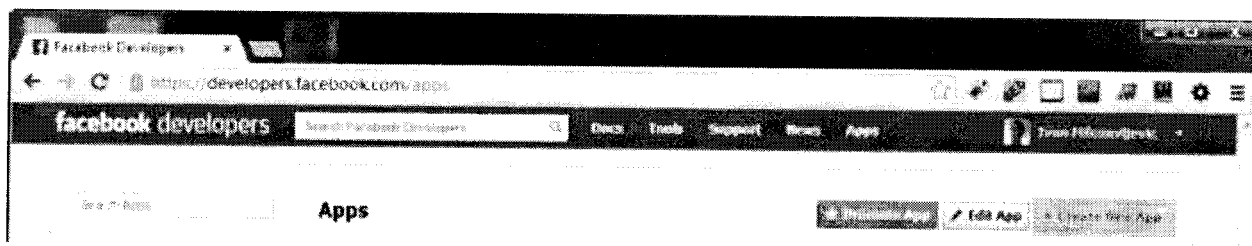
Приликом регистрације потребно је пратити кораке регистрације (слика 25).



Слика 35: Кораци при регистрацији Facebook developer

Креирање Facebook апликације

За креирање апликације у домену друштвене мреже Facebook потребно је да корисник буде регистрован као *developer* на овај друштвеној мрежи. Креирање може се започети доласком на веб адресу: <https://developers.facebook.com/apps>.



Слика 36: Креирање апликације на друштвеној мрежи *Facebook*

Апликације које могу да се креирају на овој друштвеној мрежи могу бити за различите платформе и то:

- Веб апликације
- *Android* апликације
- *iOS* апликације



Слика 37: Платформе за развој апликација на друштвеној мрежи *Facebook*

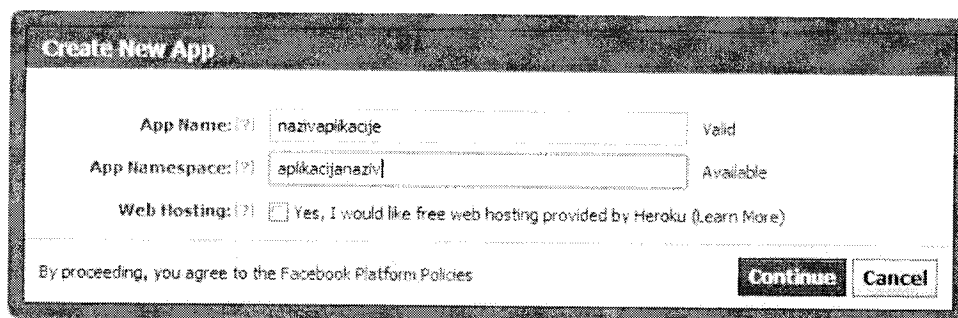
Једна апликација може да се направи за све три платформе. Апликација за тестирање математичког знања у домену *Facebook* друштвене мреже је направљена је за једну платформу и то за веб платформу.

Функције које је могуће искористити у домену друштвене мреже *Facebook* налазе се у званичним библиотекама ове друштвене мреже, односно налазе се у скупу за развој апликација који се назива *Software Development Kit (SDK)* [64]. За развој апликација на вебу друштвена мрежа *Facebook* развија свој званичан *SDK*. У апликацији за тестирање математичког знања користе се званични *SDK* за програмске језике *JavaScript* и *PHP*.

Развоја апликације захтева испуњавање одређених техничких услова. Потребно је поседовати веб сервер и то са:

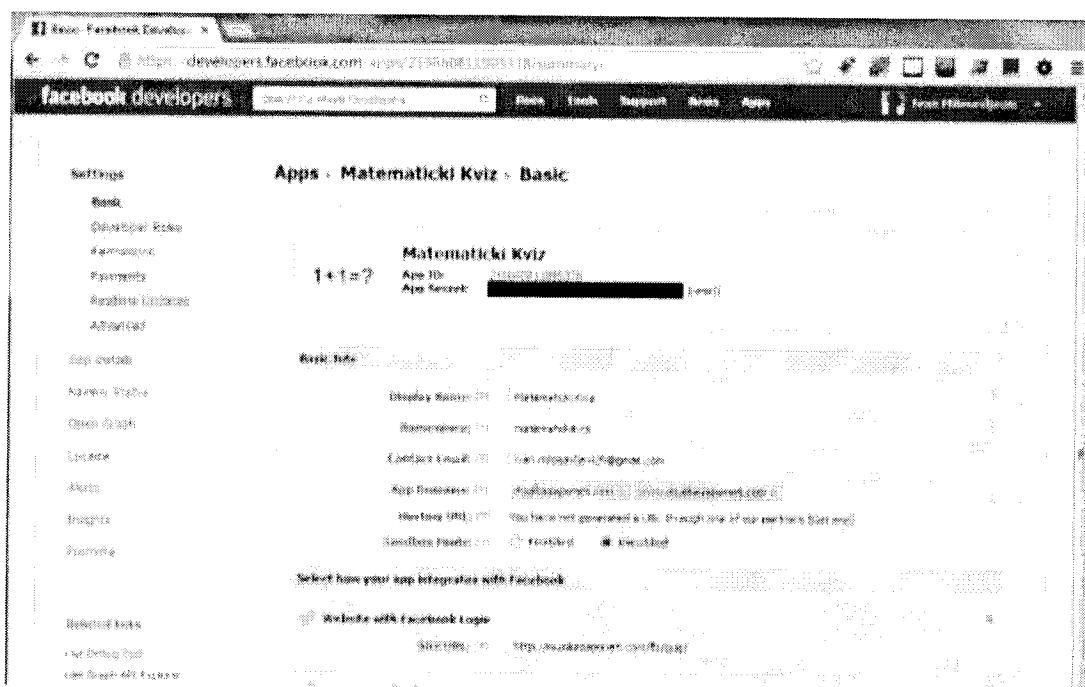
- *SSL* [65] сертификатом,
- *PHP* верзије 5+.

Уколико постоје технички услови онда је могуће креирати апликацију (слика 26), Први корак јесте конфигурирање апликације за веб платформу. При креирању апликације потребно је дефинисати јединствен назив апликације и назив по ком ће апликација бити лако уочљива на овој друштвеној мрежи (слика 28). Потребно је да оба назива буду валидна, односно доступна на друштвеној мрежи.



Слика 38: Дијалог за креирање нове апликације на *Facebook*-у

Након креирања апликације потребно је подесити параметре који ће омогућити развој апликације и коришћење дефинисаних *SDK* (слика 24).



Слика 39: Подешавање апликације на *Facebook*-у

Уколико су подешавања апликације коректна развој апликације је могућ.